

CURSO PRACTICO DEL ORDENADOR PERSONAL, EL MICRO Y EL MINIORDENADOR

Editorial Delta, S.A.

MI COMPUTER CURSO PRACTICO
DEL ORDENADOR PERSONAL, EL MICRO Y EL MINIORDENADOR

Publicado por Editorial Delta, S.A., Barcelona, y comercializado en exclusiva por Distribuidora Olimpia, S.A., Barcelona

Volumen III - Fascículo 31

Director: José Mas Godayol
Director editorial: Gerardo Romero
Jefe de redacción: Pablo Parra
Coordinación editorial: Jaime Mardones, Francisco Martín
Asesor técnico: Jesús Nebra

Redactores y colaboradores: G. Jefferson, R. Ford, S. Tarditti, A. Cuevas, F. Blasco
Para la edición inglesa: R. Pawson (editor), D. Tebbutt (consultant editor), C. Cooper (executive editor), D. Whelan (art editor), Bunch Partworks Ltd. (proyecto y realización)

Realización gráfica: Luis F. Balaguer

Redacción y administración:
Paseo de Gracia, 88, 5.°, Barcelona-8
Tels. (93) 215 10 32 / (93) 215 10 50 - Télex 97848 EDLTE

MI COMPUTER, *Curso práctico del ordenador personal, el micro y el miniordenador*, se publica en forma de 96 fascículos de aparición semanal, encuadernables en ocho volúmenes. Cada fascículo consta de 20 páginas interiores y sus correspondientes cubiertas. Con el fascículo que completa cada uno de los volúmenes, se ponen a la venta las tapas para su encuadernación.
El editor se reserva el derecho de modificar el precio de venta del fascículo en el transcurso de la obra, si las circunstancias del mercado así lo exigieran.

ISBN: 84-85822-83-8 (fascículo) 84-85822-94-3 (tomo 3)
84-85822-82-X (obra completa)
Depósito Legal: B. 52/1984

Fotocomposición: Tecfa, S.A., Pedro IV, 160, Barcelona-5
Impresión: Cayfosa, Santa Perpètua de Mogoda (Barcelona) 158408
Impreso en España - Printed in Spain - Agosto 1984

Editorial Delta, S.A., garantiza la publicación de todos los fascículos que componen esta obra.

Distribuye para España: Marco Ibérica, Distribución de Ediciones, S.A., Carretera de Irún, km 13,350. Variante de Fuencarral, Madrid-34.
Distribuye para Argentina: Viscontea Distribuidora, S.C.A., La Rioja 1134/56, Buenos Aires.
Distribuye para Colombia: Distribuidoras Unidas, Ltda., Transversal 93, n.° 52-03, Bogotá D.E.
Distribuye para México: Distribuidora Intermex, S.A., Lucio Blanco, n.° 435, Col. San Juan Tlihuaca, Azcapotzalco, 02400, México D.F.
Distribuye para Venezuela: Distribuidora Continental, S.A., Ferrenquín a Cruz de Candelaria, 178, Caracas, y todas sus sucursales en el interior del país.

Pida a su proveedor habitual que le reserve un ejemplar de MI COMPUTER. Comprando su fascículo todas las semanas y en el mismo quiosco o librería, Vd. conseguirá un servicio más rápido, pues nos permite realizar la distribución a los puntos de venta con la mayor precisión.

Servicio de suscripciones y atrasados (sólo para España)

Las condiciones de suscripción a la obra completa (96 fascículos más las tapas, guardas y transferibles para la confección de los 8 volúmenes) son las siguientes:

a) Un pago único anticipado de 16 690 ptas. o bien 8 pagos trimestrales anticipados y consecutivos de 2 087 ptas. (sin gastos de envío).

b) Los pagos pueden hacerse efectivos mediante ingreso en la cuenta 3371872 de la Caja Postal de Ahorros y remitiendo a continuación el resguardo o su fotocopia a Distribuidora Olimpia (Paseo de Gracia, 88, 5.°, Barcelona-8), o también con talón bancario remitido a la misma dirección.

c) Se realizará un envío cada 12 semanas, compuesto de 12 fascículos y las tapas para encuadernarlos.

Los fascículos atrasados pueden adquirirse en el quiosco o librería habitual. También pueden recibirse por correo, con incremento del coste de envío, haciendo llegar su importe a Distribuidora Olimpia, en la forma establecida en el apartado b).

Para cualquier aclaración, telefonear al (93) 215 75 21.

No se efectúan envíos contra reembolso.

Con este fascículo se incluyen las portadillas correspondientes al segundo volumen (fascículos 13-24)



Sistemas de gestión comercial

Uno de los sistemas de gestión más sencillos consiste en reflejar tan sólo el movimiento de caja del negocio

Son tres los programas que hemos elegido para ilustrar las exigencias de diseño de un paquete de gestión: *Book-keeping system for the cash trader* (sistema de teneduría de libros para la compra-venta al contado) de Quick-Count, un programa grabado en cinta para el Commodore 64; *Accountant* (el contable), de Compact Accounting Services, pensado para el BBC Modelo B Micro equipado con unidades de disco y el segundo procesador Z80 de Acorn; y *Microledger* (microlibro mayor), de Lewis Ashley Computers, un sistema en disco que funciona en el Apple II.

Si es que ha de satisfacer las exigencias de teneduría de libros, un programa ha de ser capaz de distinguir las diversas fuentes de ingresos y gastos del negocio. El negociante no se conforma con tener un resumen global de cuánto se ha hecho de caja o cuánto se ha gastado en un período dado. Necesita analizar los totales disponiendo de cada uno de los conceptos que los integran. Necesita saber cuánto se le fue en alquiler, en viajes, en pequeños gastos, en papelería y otras partidas similares. Si así no fuera, la única herramienta contable que necesitaría el comerciante sería una máquina de sumar las entradas y las salidas.

Para satisfacer esta necesidad de análisis del movimiento del dinero, el programa ha de dar al usuario capacidad para apuntar los ingresos y los gastos según un "plan de cuentas". El *libro mayor* es un instrumento contable donde se anotan tales cuentas. Constituye el corazón de todo sistema de gestión comercial.

Cash trader nos servirá de excelente ejemplo para mostrar tanto lo que se puede conseguir con un programa en cassette como las limitaciones de esta clase de programas. Consta de un libro mayor que contiene 79 cuentas distintas. Son muy pocas, en comparación con los sistemas basados en disco, el más pequeño de los cuales ofrece centenares de cuentas individuales, pero suelen ser más que suficientes para un gran número de comerciantes que sólo desean identificar los conceptos de mayor interés en su negocio.

El libro mayor de *Cash trader* viene con "estructura previa" (a diferencia tanto del *Accountant* como del *Microledger*, que son libros mayores de formato libre). Esto significa que la codificción numérica de las cuentas está ya establecida en dicho libro. Esto hace que el programa sea más sencillo a la hora de decidir qué operaciones ha de llevar a cabo con los datos de cada cuenta al objeto de elaborar los distintos informes. Facilita además su puesta en marcha por el usuario. Aunque, por otra parte, impone restricciones para una libre y más apropiada reestructuración del libro mayor.

Los números del 01 al 19, por ejemplo, son para las cuentas de ingresos. Por mejor decirlo, se trata de los apuntes provenientes de ventas y otros ingresos (incluidas las mercancías en stock) a favor de la empresa. Sin embargo, puede que el comerciante sólo desee utilizar cuatro de estas 19 posibles cuentas. Una cuenta individual, pongamos por caso la número 01, se podría denominar "Ingresos", y dar entrada (el término técnico es "pasar") a todos los ingresos generados por el comercio. La cuenta 01 llevaría entonces un total acumulativo de todos los ingresos.

Por otra parte, puede que al comerciante le interese obtener saldos separados para, supongamos, cinco categorías distintas de mercancías vendidas. Esto significa la apertura de cinco cuentas de ingresos (del número 01 al 05). Entonces sería necesario preparar procedimientos para asegurar que cada venta se pasa a su cuenta. *Cash trader* no tendría ninguna dificultad con este detalle.

Marcus Wilson-Smith

Minoristas
Un micro personal y una cassette pueden bastar para el funcionamiento de un pequeño negocio. La mayoría de los paquetes de este nivel tienen un modesto objetivo, como por ejemplo tratar sólo con el flujo de caja y no con sistemas completos de contabilidad

Los números del 20 al 49 se reservan para las cuentas de compras y otros gastos. Esta sección está diseñada para cubrir los distintos tipos de gastos a que ha de hacer frente la empresa. El comerciante empleará nombres de cuentas como: alquiler, sueldos, impuestos, comisiones bancarias, intereses de préstamos, material de oficina, publicidad, teléfono.

Los números del 50 al 79 corresponden a cuentas de balance. Éstas reflejan el estado financiero global de la empresa en cualquier momento dado. Nombres de cuentas de balance son, por ejemplo, la cuenta de inmovilizado (que muestra el valor de todos los bienes fijos que posee la empresa), la cuenta bancos (o las cuentas, si hubiera más de una), la cuenta del IVA (impuesto al valor añadido), la cuenta acreedores (que muestra lo que adeuda la empresa en concepto de compras y gastos a crédito).

En un sistema de contabilidad manual, el comerciante habría de anotar sus operaciones en los libros de ventas y de compras, y las sumas totales reflejarían los gastos e ingresos del día o de la semana. El menú *Cash trader* para pasar los saldos al libro mayor casi se explica por sí solo:

1. Ingresos diarios
2. Pagos en efectivo
3. Pagos por banco
4. Sueldos

La opción 1, Ingresos diarios, está diseñada para registrar los ingresos de una semana. Le solicita al usuario que seleccione un día (numerados del 1 al 7) y luego que dé entrada al total de ingresos de ese día. La única distinción posible con las ventas es la que permite al usuario señalar un total, mediante una bandera, como una "suma especial".

Así se atiende al hecho de que no todos los ingresos provienen de ventas ordinarias. Si una tienda de jardinería vende su furgoneta, obtiene un ingreso extraordinario, y englobar ese efectivo en los ingresos por ventas del día produciría una cifra de ventas distorsionada. El *Cash trader*, por sorprendente que parezca, tratándose de un programa tan sencillo, posee las facilidades necesarias para identificar las ventas "especiales". Pero éste es el único análisis o detalle del total de ventas diarias que permite el programa. El usuario puede optar por pasar la cantidad a una de tres cuentas nominales: la cuenta de bancos, la de caja o la cuenta transitoria de tarjeta de crédito.

En cuanto a los pagos (el equivalente al libro de compras), el sistema permite que el usuario pague ya sea mediante la cuenta de caja o la cuenta de bancos. Un código de referencia de tres dígitos identifica cada una de las operaciones de pago. Y permite una descripción narrativa de 16 caracteres del motivo de cada desembolso (por lo general, el nombre del proveedor). Asimismo, calcula el IVA repercutido sobre los pagos y pasa la cantidad a la cuenta de control del IVA.

Todo aquel que utilice el *Microledger* o el *Accountant* habrá de organizar un plan contable similar. No obstante, entre ambos existe una gran diferencia dado que estos dos sistemas carecen de un orden predefinido para las cuentas. Con el *Accountant*, por ejemplo, se le otorga al usuario una estructura de códigos de ocho dígitos y éste puede hacer corresponder cualquier cuenta con los números que guste, cualesquiera sean. El *Mocroledger* posee una estructura de códigos de tres dígitos, pero se aplica el principio de formato libre.

El programa *Accountant* también se ha elaborado en torno a un libro mayor general, aunque mucho más sofisticado, capaz de tratar con muchas más cuentas. También dispone de una facilidad que posibilita el análisis. Pero obliga al usuario a resumir hasta cierto punto las entradas de datos, en comparación con un sistema completo de libro de ventas y de compras.

Con un programa completo de libro de ventas y libro de compras, el usuario tiene los medios para llevar un archivo maestro de los clientes y de los proveedores, respectivamente. Este archivo incluirá detalles completos de la cuenta del cliente o del proveedor y también llevará un registro completo de todas las transacciones más sobresalientes llevadas a cabo con ese cliente o ese proveedor. El usuario tendrá la posibilidad de llamar la cuenta a la pantalla y disponer de un listado completo de todas las facturas pasadas a esa cuenta y de todos los recibos de pagos efectuados en dicha cuenta.

El *Accountant* no llega a proporcionar estos medios. En cambio, al igual que el *Cash trader*, asume el enfoque del apunte diario. Pero como es un sistema en disco que dispone de un espacio de memoria considerablemente mayor, no exige que la información tenga una entrada de forma tan condensada.

En lugar de dar entrada a una sola cifra por día de la semana, el usuario puede realizar todas las entradas que sean necesarias. El sistema reconoce cinco tipos distintos de operaciones: facturas, créditos concedidos, recibos de caja, ventas al contado y recibos varios. Cada una de las entradas se puede describir con un máximo de 16 caracteres, se les puede otorgar un número de referencia exclusivo y se las puede analizar en cualquier número de cuentas del libro mayor general (en vez de la simple opción de tres posibilidades que ofrece *Cash trader*).

El libro de compras puede ocuparse de las compras a crédito, compras al contado y otras compras. También posee una facilidad para informar detalladamente sobre el IVA.

El *Microledger*, a diferencia de estos dos sistemas, lleva un libro mayor completo de compras y ventas. El usuario puede crear hasta 999 cuentas individuales del libro mayor de compras y ventas. Cada cuenta registrará el nombre del cliente y hasta cinco líneas de dirección. La cuenta mantiene de forma automática un balance acumulativo y una lista de todas las operaciones más relevantes.

La principal diferencia existente entre el *Microledger* y los otros dos paquetes que hemos descrito previamente, es que éste contiene una gran cantidad de información sobre los negocios efectuados con cada cliente y con cada proveedor. El *Cash trader* condensa esa información en una o dos sumas totales. El *Accountant* puede procesar transacciones individuales con clientes o proveedores, pero el usuario necesitará llevar un libro de ventas o compras manual con el objeto de ver con facilidad cuál es el saldo de cada cliente o proveedor.

En el próximo capítulo de esta serie continuaremos con la comparación entre estos tres paquetes, analizando las formas en que manipulan la entrada de valores en los títulos de cuentas nominales.

Justificantes
Arthur está haciendo sus cuentas de la semana. Ha vendido tres cajas de whisky Glen Kyushu al contado, y 10 cajas mediante tarjeta de crédito; ha recibido un talón de 500 libras esterlinas por servicios prestados y ha vendido el Rolls Royce en efectivo. Sus pagos de la semana han sido la compra al contado de más whisky, pagarle en efectivo a Terry, su único empleado, y un talón para comprarse un coche nuevo

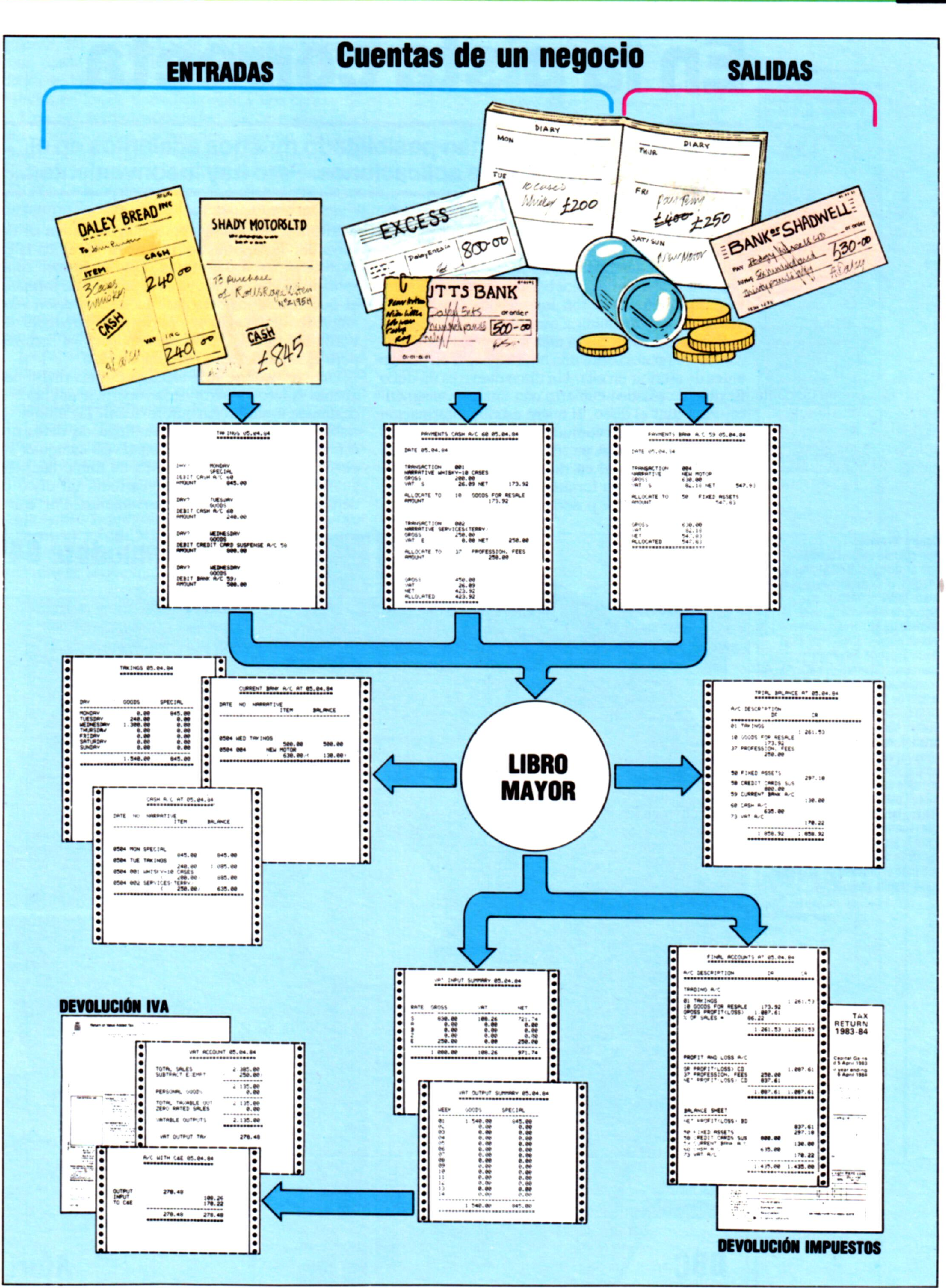
Entrada de cobros
A todos los ingresos se les da entrada de esta forma, y se marcan como ARTÍCULOS o ESPECIAL a tenor de la naturaleza de la venta, y EFECTIVO, BANCO o TARJETA DE CRÉDITO según la forma en que haya pagado el cliente

Entrada de pagos
A todos los pagos se les da entrada de la misma forma, indicando EFECTIVO o BANCO, la parte del IVA y el número de código de la cuenta del libro mayor que recogerá la cantidad

Informes
Estos resúmenes se generan automáticamente una vez finalizada la entrada COBROS o PAGOS

Libro mayor
Consta de hasta 79 cuentas nominales tituladas: COBROS, STOCK, SALARIOS, ALQUILERES, CARGOS BANCARIOS, ACTIVOS FIJOS, CAJA, etc. Éstas, al objeto del balance, están agrupadas en la cuenta de COMPRA Y VENTA, la cuenta de PÉRDIDAS Y GANANCIAS y la HOJA DE BALANCE

Informes
Se puede disponer de diversos informes: el estado de la cuenta de CAJA y la cuenta BANCARIA y un BALANCE DE COMPROBACIÓN son lo que la mayoría de los comerciantes desearía ver después de dar entrada a las operaciones de una semana o de un día. La cuenta FINAL y el informe del IVA probablemente sólo se requerirán una vez por trimestre


Cuentas de un negocio
ENTRADAS
SALIDAS
DALEY BREAD INC
SHADY MOTORS LTD
EXCESS
BANK OF SHADWELL
DIARY
LIBRO MAYOR
DEVOLUCIÓN IVA
DEVOLUCIÓN IMPUESTOS
TAX RETURN 1983-84
Tony Duncan Smith

En la pista correcta

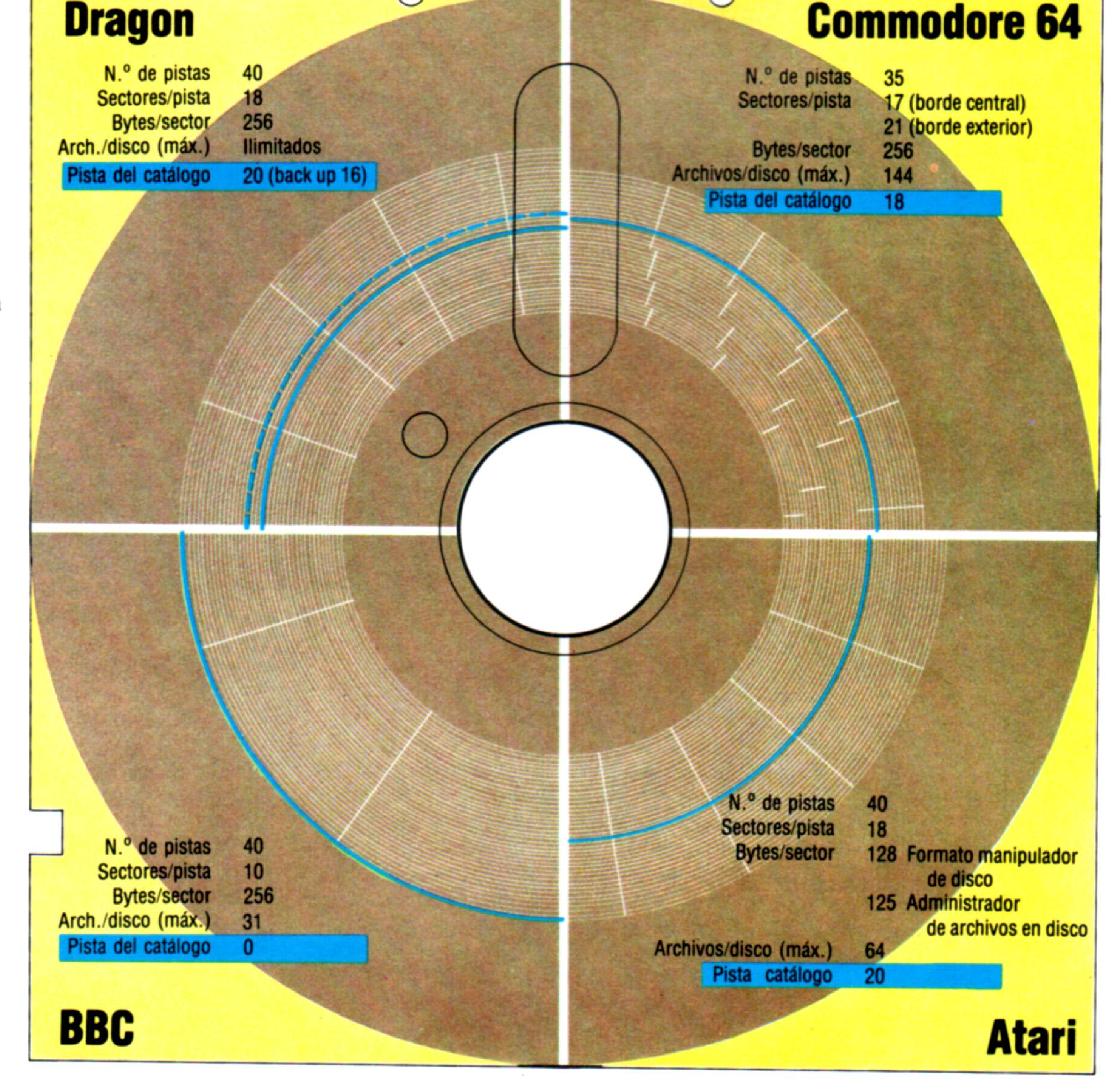
Las unidades de disco han posibilitado muchos adelantos en el campo del software para aplicaciones. Pero hay inconvenientes...

Cuando usted adquiere una caja de discos nuevos, no los puede utilizar de inmediato. Antes de usarlos debe formatearlos para su micro particular y, a pesar de que los principios básicos del procedimiento para dar formato son los mismos, los detalles varían de una máquina a otra.

Formatear es en cierta manera algo así como trazar los márgenes sobre una hoja de papel en blanco antes de escribir en ella. Un disco nuevo es un disco flexible de plástico cubierto con una caja magnética. Para usar el disco, el micro escribe información que lo divide en un conjunto de pistas concéntricas y que subdivide éstas en sectores como si se tratara de "cortar un pastel en raciones". Algunas unidades de disco dan un formato de 40 pistas de datos, otras de 80. Algunas poseen dos cabezas de lectura-escritura y, por consiguiente, dan formato a las dos caras del disco, otras sólo escriben y leen en la superficie superior del mismo. Esto se conoce como *sectorización soft*, porque es el proceso de formateo el que marca los sectores. La *sectorización hard* empleaba unos agujeros perforados alrededor del borde interior del disco para marcar los sectores, pero esta técnica casi ha desaparecido.

Una vez que se han efectuado estas divisiones, apenas la tercera parte de la superficie del disco es destinada a almacenar información. La última variable que incide sobre la cantidad de datos que caben en un disco es la *densidad*. En cualquier superficie determinada, un disco de doble densidad guarda el doble de lo que guardaría un disco de densidad simple, y así sucesivamente. Por consi-

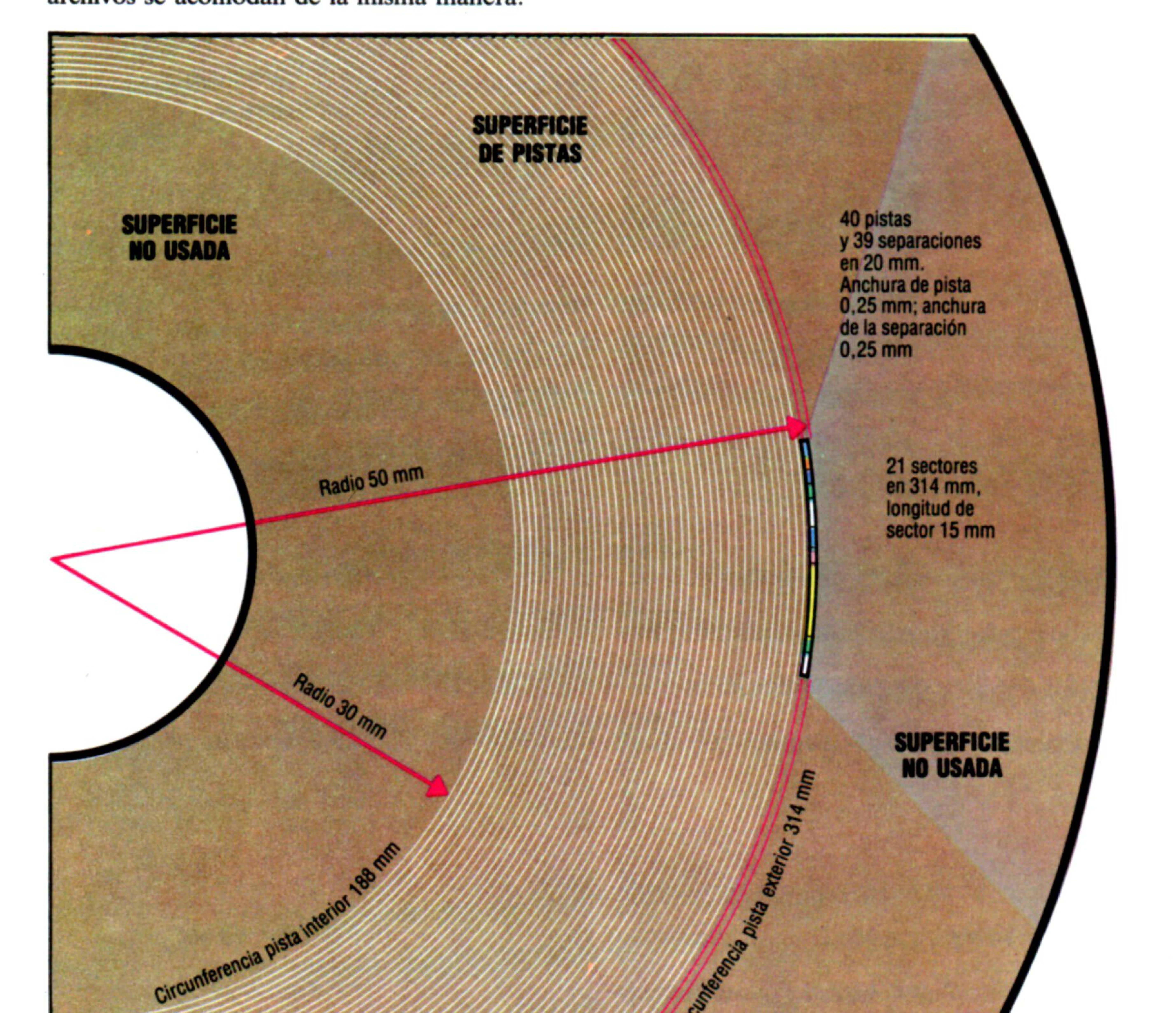
Cuatro formatos
Aun cuando utilizan el mismo tipo de disco, el BBC Micro, el Dragon, el Commodore 64 y el Atari, todos formatean sus discos de diferente manera. El número de archivos separados que se pueden almacenar en el disco depende del sistema operativo. Observe que el posicionamiento en el disco de la lista de archivos del DOS (la pista del catálogo), en un disco BBC se halla en el borde exterior, mientras que en las otras máquinas está en una pista central. Se suele preferir la posición central, ya que la cabeza del disco, por término medio, tendrá menos distancia que recorrer cuando se desplace entre la pista del catálogo y las pistas de datos reales, haciendo que todo el proceso de acceso al disco resulte más rápido



guiente, las capacidades de los discos flexibles varían, desde unos 90 Kbytes en el caso de un disco de densidad simple de una sola cara, hasta 1,2 Mbytes en los de densidad doble y dos caras.

Una vez formateado, usted puede copiar en el disco programas e información. Esto no se realiza directamente. Todo se conserva en archivos y se controla mediante el sistema operativo en disco (DOS: *Disk Operating System*). Éste lleva en un directorio la lista de los archivos que contiene el disco y utiliza entonces los sectores libres que resulten convenientes para almacenar la información. Físicamente, un archivo puede estar esparcido a través de todo el disco, y el DOS también llevará un índice de cuáles son los sectores que ocupó un archivo, dejando señaladores en cada sector indicando dónde se halla la siguiente parte del archivo.

La capacidad de tomar un disco de una máquina y utilizarlo con otra depende de dos cosas: de cómo sea el formato del disco y de cuántos archivos estén acomodados en el mismo. Por suerte, en la actualidad muchos micros emplean el mismo formato de disco o, al menos, poseen programas de utilidades especiales que pueden leer y escribir formatos extraños. Muchas máquinas utilizan sistemas operativos estándar como el CP/M (*Control Program Microprocessors*: programa de control para microprocesadores) y MS-DOS (*Microsoft Disk Operating System*: sistema operativo en disco Microsoft), y los archivos se acomodan de la misma manera.



Pistas y sectores
En el disco los datos se escriben en pistas concéntricas. Cada pista está dividida en sectores. Un sector contiene un bloque de los datos del usuario, y a este bloque el DOS le adjunta automáticamente los datos de verificación de errores y de identificación necesarios para la administración de archivos y registros

ENCABEZAMIENTO DEL SECTOR

Señal de sincronización
Permite emparejar la velocidad de lectura de la cabeza del disco con la velocidad de rotación

Número de pista
Identifica la pista en curso

Número de sector
Identifica el sector en curso

Suma de control 1
Proporciona una verificación de errores de los datos del encabezamiento

Separación 1
Deja el encabezamiento del sector aparte del bloque de datos

BLOQUE DE DATOS DEL SECTOR

Señal de sincronización 2

Enlace de archivo
Identifica el siguiente sector del archivo en curso

Bloque de datos
128, 256 o 512 bytes de datos del usuario, según el sistema operativo en disco

Suma de control 2
Proporciona una verificación de errores de los datos del usuario

Separación 2
Separa este sector del siguiente

La respuesta lógica

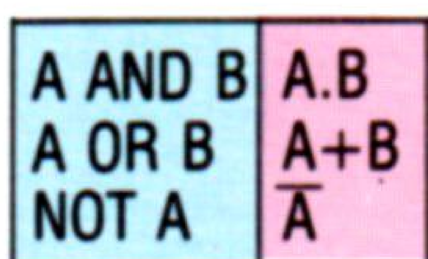
Esta vez haremos alto en nuestro curso y resumiremos las reglas de lógica y álgebra booleanas estudiadas hasta ahora

Vamos a efectuar una breve revisión del trabajo que hemos realizado en las lecciones anteriores. Los principios de la lógica se aplican a los ordenadores en el diseño del hardware que debe llevar a cabo ciertas operaciones especiales. Ya hemos diseñado un circuito sumador, el cual al combinarse con otros circuitos sumadores permite la adición de números binarios (véase p. 528). Este circuito reproduce el método tradicional de la suma, permitiendo llevar dígitos de una columna a la siguiente.

En el diseño de este circuito utilizamos tres elementos básicos, denominados puertas lógicas. Las funciones que podían realizar estas puertas se indicaban mediante los nombres que les habíamos asignado (AND, OR y NOT), y definimos cada una de ellas mediante una tabla de verdad. La tabla de verdad es una forma sencilla de reflejar por escrito la(s) salida(s) de un circuito para cualquier posible combinación de entradas. Con dos entradas son posibles cuatro (2^2) combinaciones de entradas, con tres entradas se hacen ocho (2^3) combinaciones, y así sucesivamente. Asimismo, vimos cómo las puertas lógicas se pueden enlazar entre sí para producir ciertas salidas deseadas. Estos circuitos más complicados también se podían describir mediante sus tablas de verdad. En particular, diseñamos un circuito OR, exclusivo de cinco puertas, que nos dio una salida verdadera sólo cuando *una* de sus entradas era verdadera (véase p. 512).

Álgebra booleana

La combinación de los elementos lógicos se puede describir sobre el papel mediante un conjunto de símbolos muy parecidos a los del álgebra normal. La rama de las matemáticas que se ocupa de la representación de la lógica se denomina álgebra booleana, en honor del matemático inglés George Boole (1815-1864), que fue quien definió por primera vez sus principios. Cada uno de los tres elementos básicos de la lógica posee su propio símbolo especial:



A AND B	A.B
A OR B	A+B
NOT A	$\overline{A}$

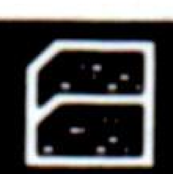
Así como existen leyes que gobiernan el manejo de cifras en aritmética y de letras en álgebra, del mismo modo existen leyes especiales que gobiernan la simplificación de las expresiones lógicas. Las leyes del álgebra booleana están resumidas en la tabla siguiente.

RELACIONES ESPECIALES	
Relación	**Dual**
A.A=A	A+A=A
$A.\overline{A}=0$	$A+\overline{A}=1$
A.0=0	A+1=1
A.1=A	A+0=A
A.(A+B)=A	A+A.B=A
$A.(\overline{A}+B)=A.B$	$A+\overline{A}.B=A+B$
LEYES DE MORGAN	
1) $\overline{A+B}=\overline{A}.\overline{B}$	
2) $\overline{A.B}=\overline{A}+\overline{B}$	
PROPIEDAD ASOCIATIVA	
A.(B.C)=(A.B).C=A.B.C	
A+(B+C)=(A+B)+C=A+B+C	
PROPIEDAD CONMUTATIVA	
A.B=B.A	
A+B=B+A	
PROPIEDAD DISTRIBUTIVA	
A.(B+C)=A.B+A.C	

Utilizando estas reglas se puede simplificar expresiones lógicas y reducir el número de puertas que requiera el circuito final. Además del método algebraico, también hemos estudiado el uso de los diagramas de Karnaugh en la simplificación de circuitos lógicos (véase p. 572). Los diagramas de Karnaugh representan un avance significativo. Aunque no nos dispensan de las simplificaciones algebraicas, reducen la cantidad de esfuerzo inherente al tratar con álgebra booleana. Estos diagramas, que en realidad son una versión de los diagramas de Venn (véase p. 526), permiten el agrupamiento de expresiones sacadas de una tabla de verdad con dos, tres y hasta cuatro variables (o sea, cuatro, ocho y dieciséis recuadros). Estos grupos representan expresiones booleanas más sencillas, de modo que es posible la simplificación. En la práctica, con frecuencia se recurrirá a una mezcla de diagramas y de álgebra para producir el circuito más eficaz.

También hemos analizado el empleo de las operaciones lógicas AND y OR en la programación en BASIC. Vimos cómo se pueden utilizar estas órdenes para combinar condiciones en sentencias IF...THEN en BASIC, y cómo le permiten al programador activar y desactivar bits aislados dentro de un registro (véase p. 546).

Como punto final de esta primera etapa del curso, hemos empleado todos nuestros conocimientos relativos a tablas de verdad, diagramas de Karnaugh, álgebra booleana y notación de puertas lógicas para diseñar circuitos que proporcionen las salidas deseadas para ciertas tareas definidas (véase p. 586). El siguiente ejercicio de revisión cubre todos los aspectos que hemos abarcado hasta el momento. Una vez que usted adquiera seguridad en el manejo de estas reglas, podremos seguir adelante para abordar la teoría lógica más avanzada.



Ejercicio de repaso

1) Un grupo pop está seguro de que su disco sencillo será un éxito si produce un buen video para acompañar a la canción y (*and*) si su casa discográfica incluye un obsequio gratis con el disco o (*or*) si aparecen en el programa de música pop más famoso de la televisión. Dibuje una tabla de verdad que refleje todos los resultados posibles. Si cada contingencia es igualmente probable, ¿qué posibilidad tiene el grupo de tener un sencillo de éxito?

2) Una escuela está organizando un club de ajedrez y se han de distribuir las tarjetas de socio. El comité decide que el número de socios se debe restringir de modo que los miembros sean:

a) Un miembro del personal;
b) Un alumno de cuarto o quinto año que estudie matemáticas y ciencias;
c) Un alumno de sexto curso que estudie matemáticas o ciencias.

Diseñe un distribuidor automático de tarjetas de socio que funcione pulsando los botones que describan mejor al posible socio. Estos botones son:

A = Alumno de cuarto curso
B = Alumno de quinto curso
C = Alumno de sexto curso
D = Miembro del personal
E = Alumno que estudia matemáticas
F = Alumno que estudia ciencias

Dibuje un diagrama del circuito para la máquina.

3) Un determinado microordenador posee un registro con la dirección 23148 (en decimal) que se utiliza para controlar la visualización en video. El bit 0 es el bit menos significativo del registro y el bit 7 es el más significativo. La pantalla se puede colocar en modalidad de alta resolución estableciendo los bits 4 y 5 en uno. Ya que los otros bits se utilizan para otras funciones, es importante no alterar sus valores. Escriba órdenes en BASIC que:

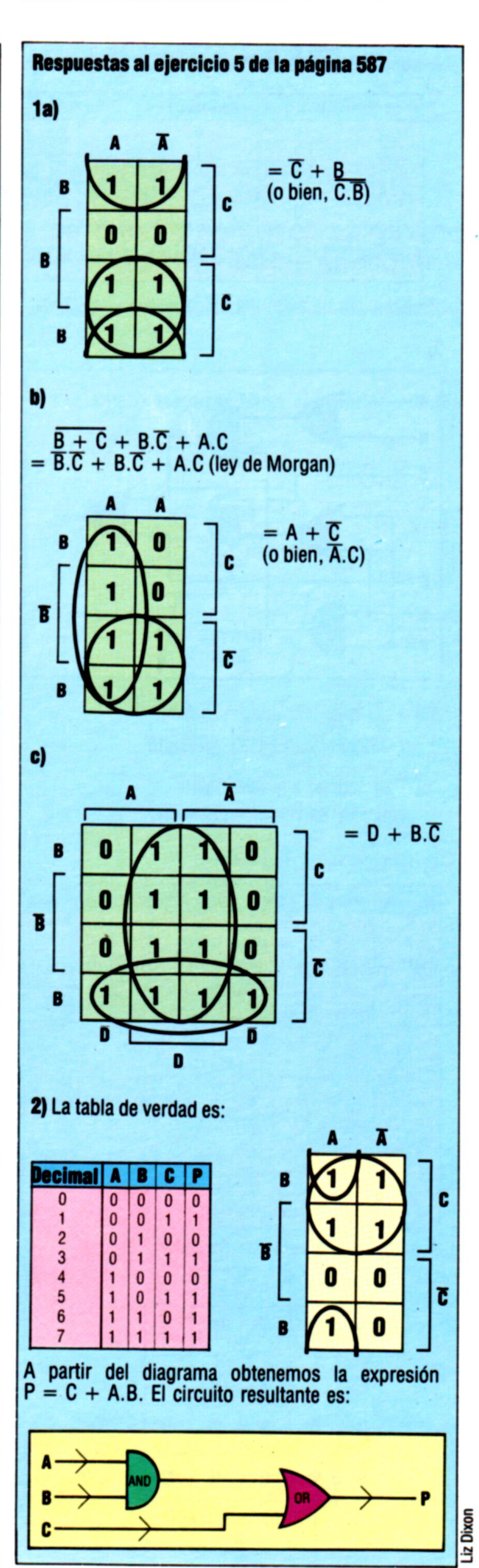
a) Enciendan la pantalla en alta resolución
b) La vuelvan a apagar

4) La cerradura de la caja fuerte de un banco se basa en interruptores operados mediante teclas. Tanto el gerente como su delegado y el cajero jefe poseen llaves. La puerta de acceso a la caja fuerte se puede abrir con dos de las tres llaves, pero:

a) Los días laborables, entre las 9 y las 17 horas, la caja fuerte sólo se puede abrir en presencia del cajero jefe.
b) En todo otro momento la caja fuerte sólo la puede abrir el gerente conjuntamente con una de las otras dos personas.

Diseñe un circuito para controlar el sistema de conmutación.

5) En un sistema digital binario de transmisión de datos, cada bit de datos alimenta simultáneamente a tres canales en paralelo que, en el extremo de recepción, alimentan a su vez un "controlador". De no haberse producido ningún error de transmisión, entonces los tres bits deberían ser iguales. En condiciones defectuosas, el controlador corregirá un error en un canal, dando una salida simple que se corresponda con la mayoría de las entradas. Diseñe un circuito para este controlador.



Respuestas al ejercicio 5 de la página 587

1a)

$= \overline{C} + B$
(o bien, $\overline{\overline{C}.\overline{B}}$)

b)

$\overline{B + C} + B.\overline{C} + A.C$
$= \overline{B}.\overline{C} + B.\overline{C} + A.C$ (ley de Morgan)

$= A + \overline{C}$
(o bien, $\overline{\overline{A}.C}$)

c)

$= D + B.\overline{C}$

2) La tabla de verdad es:

Decimal	A	B	C	P
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1

A partir del diagrama obtenemos la expresión P = C + A.B. El circuito resultante es:

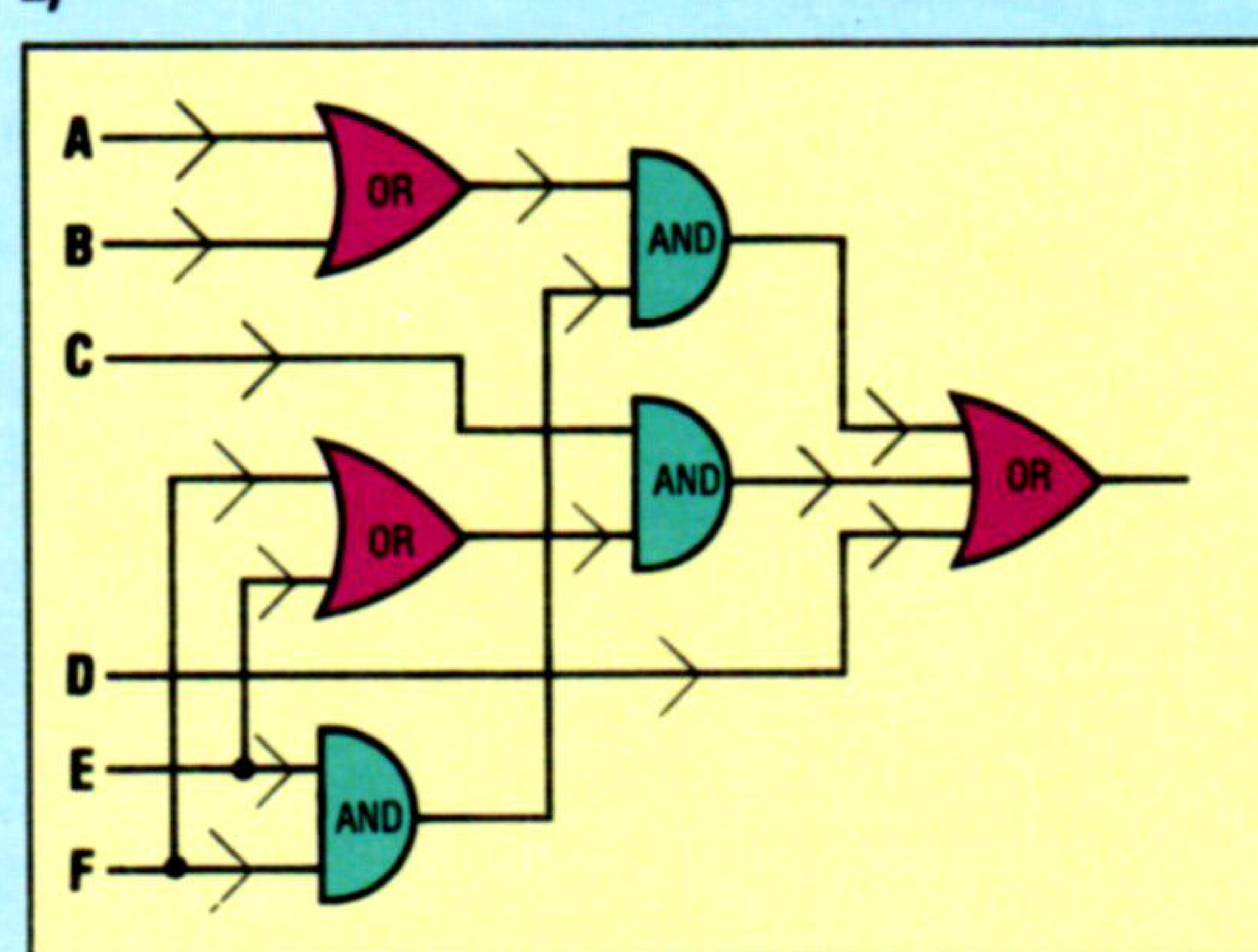
Respuestas al ejercicio de repaso

1)

Buen vídeo	Obsequio	Aparecer TV	Éxito disco
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Probabilidad de éxito del disco = **5/8 = 62,5 %**

2)



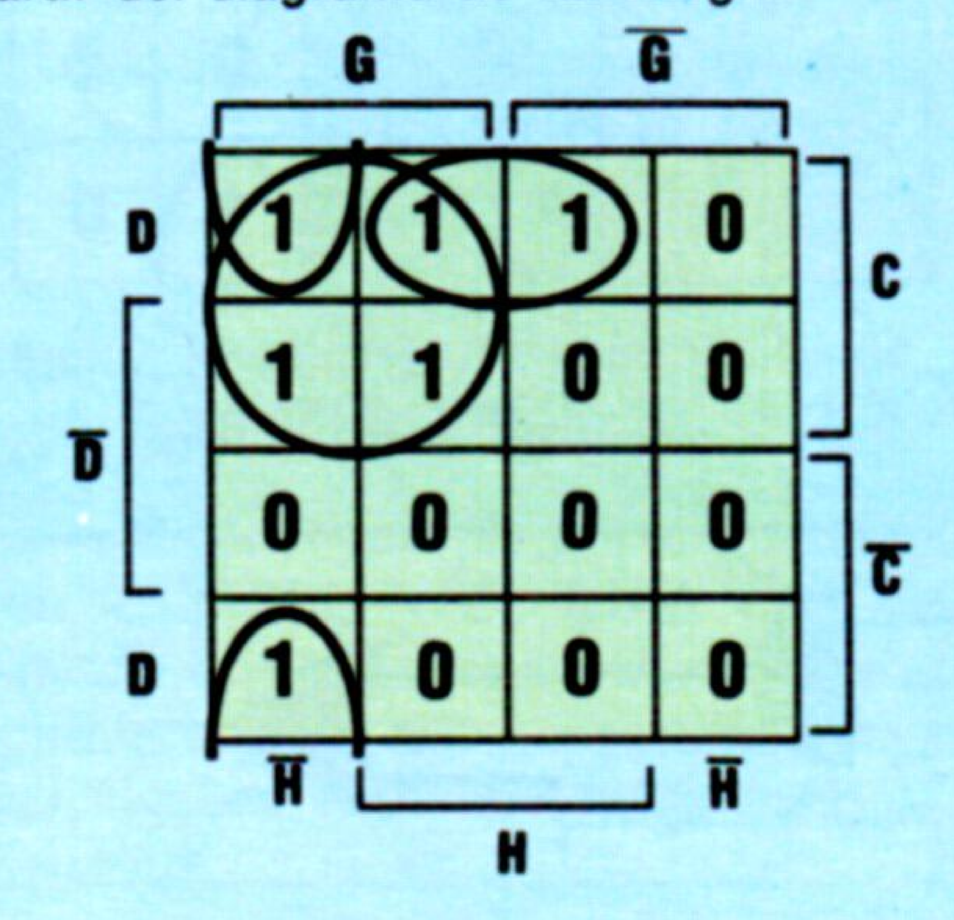
3a) Para encender alta resolución:
POKE23148,PEEK(23148)OR48

b) Para apagar alta resolución:
POKE23148,PEEK(23148)AND207

4) La tabla de verdad es:

Gerente	Delegado	Cajero jefe	9-17 horas	Abrir puerta
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	0
1	1	1	0	1
1	1	1	1	1

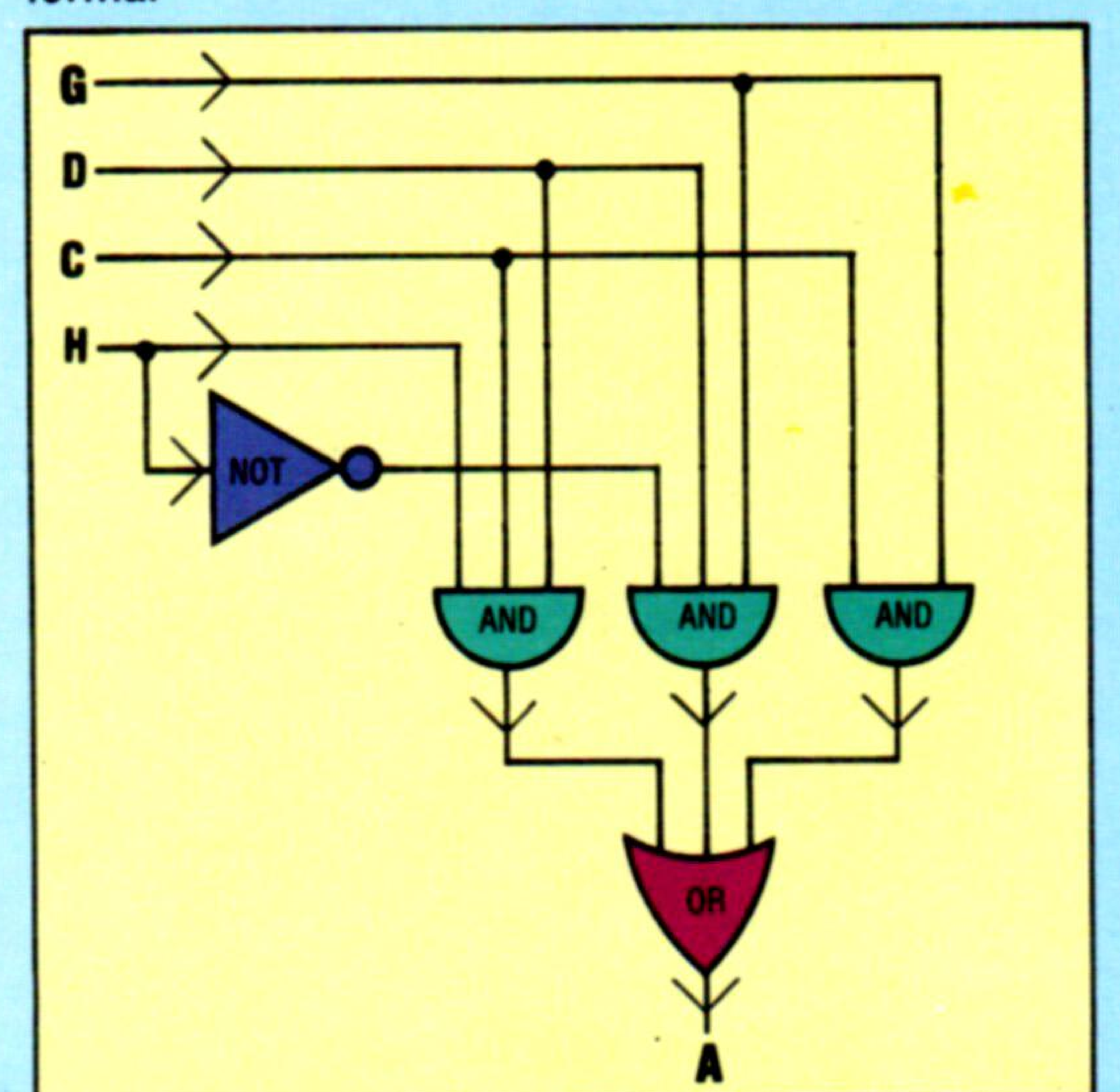
A partir del diagrama de Karnaugh:



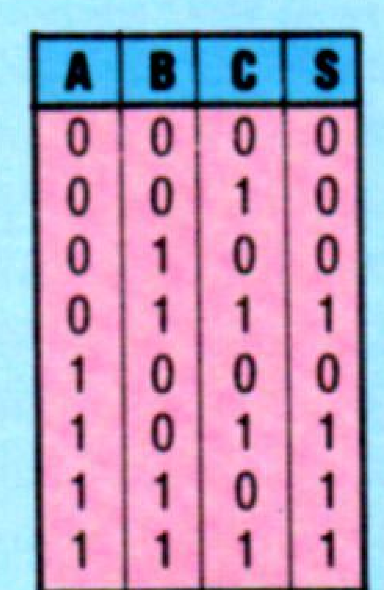
Obtenemos la expresión: $A=G.C+G.D.\overline{H}+$ $+D.C.H$. Es fácil verificar si hemos resuelto correctamente esta etapa volviendo a convertir esta expresión al idioma corriente. La expresión dice: "La puerta la pueden abrir:

a) el gerente y el cajero en cualquier momento;
b) el gerente y su delegado fuera de horario;
c) el delegado y el cajero entre las 9 y las 17 h».

El correspondiente circuito asume la siguiente forma:

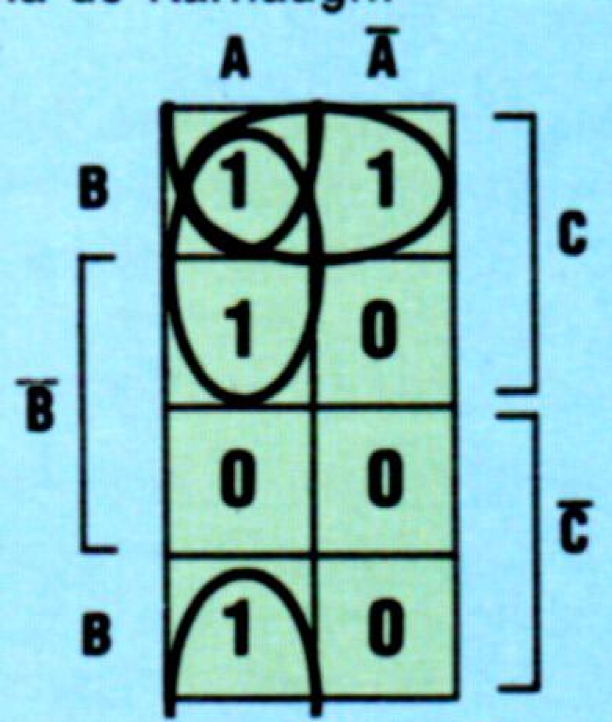


5) La tabla de verdad es:

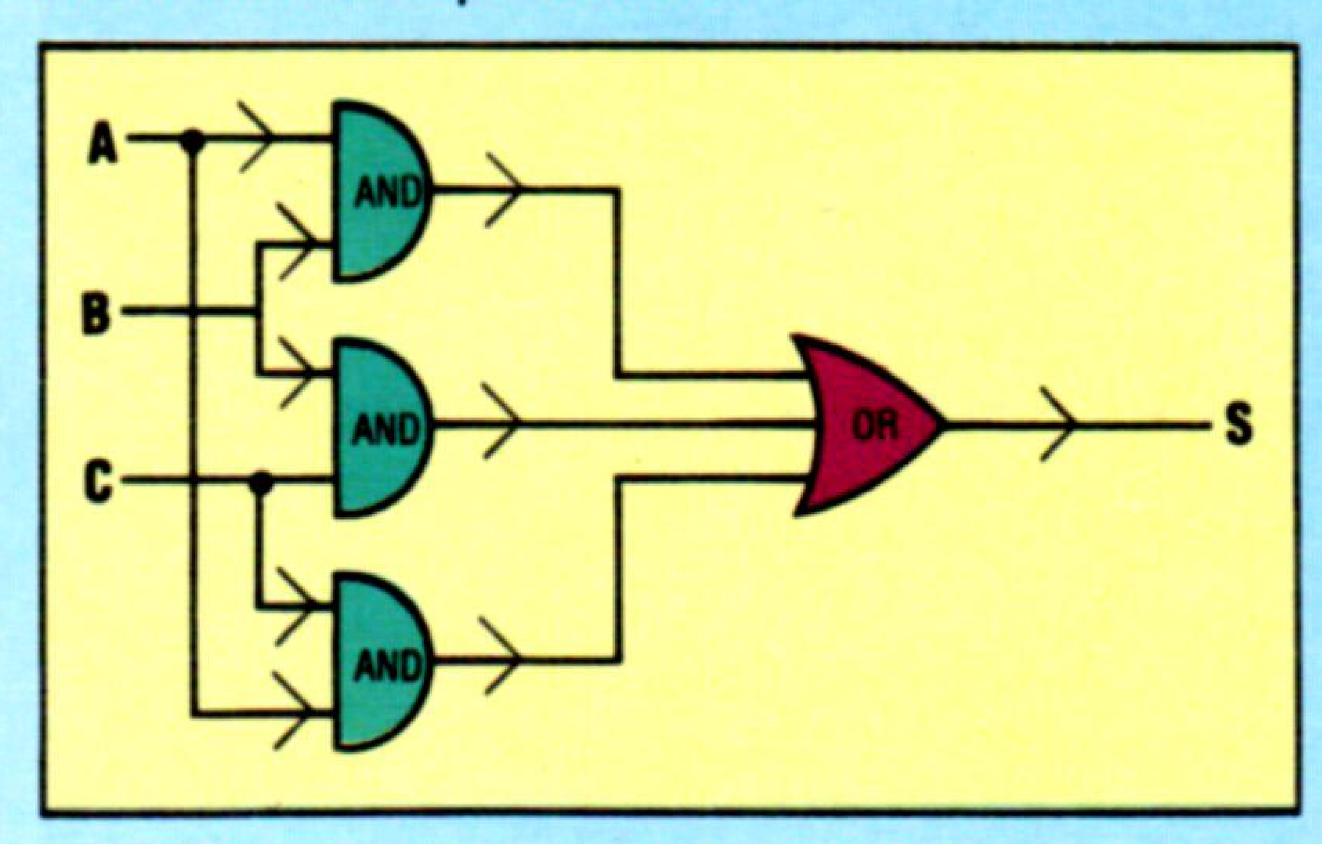


A	B	C	S
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Con el diagrama de Karnaugh:



Obtenemos la expresión: S = A.B + A.C + B.C.
El circuito correspondiente es:



Liz Dixon

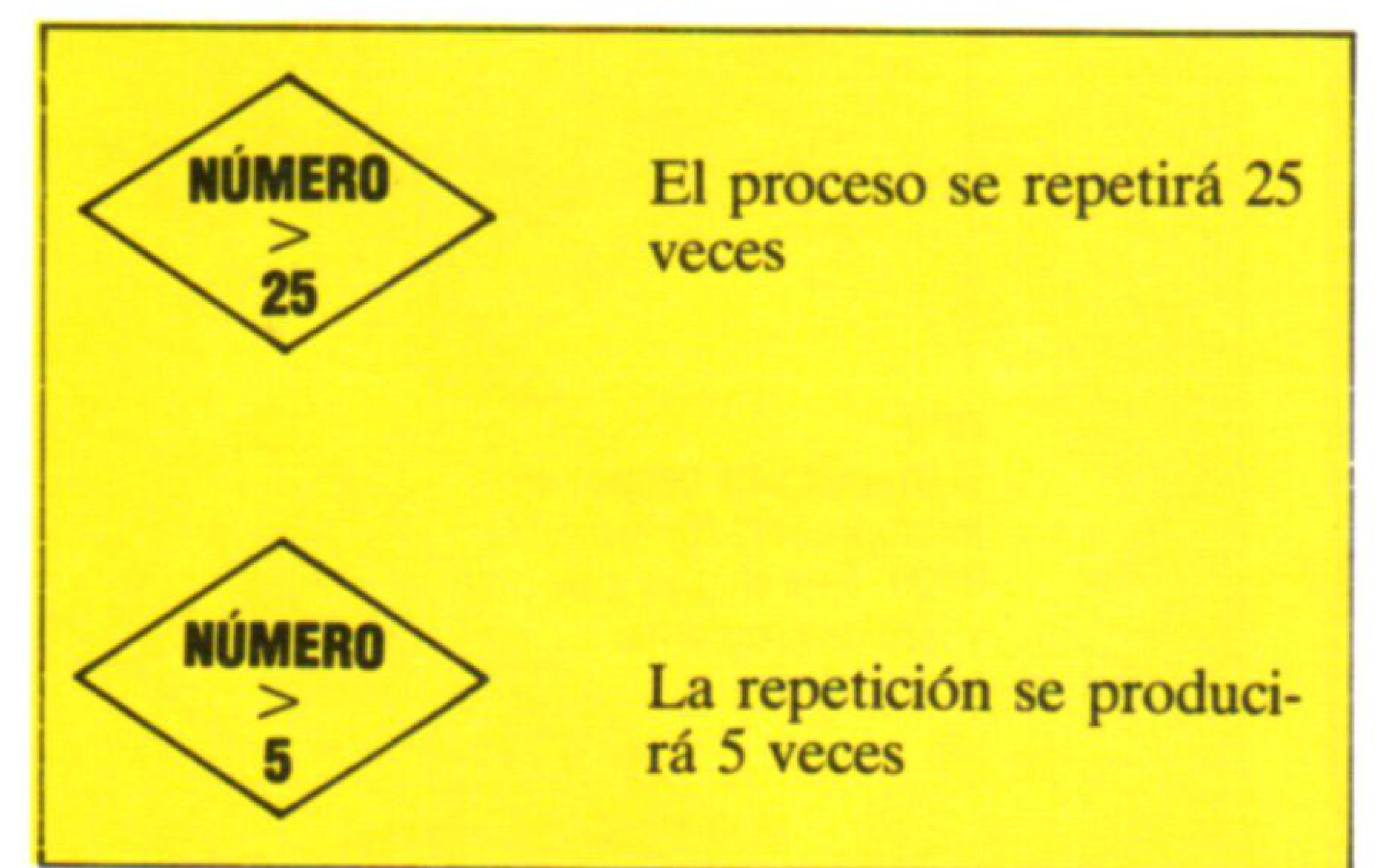
Bucles condicionados

Una noria sería un acertado símil para un bucle, cuyo movimiento tendremos que detener alguna vez

El bucle incondicionado, que vimos en el capítulo anterior, tiene unas aplicaciones limitadas. Supongamos que lo que se desea es mostrar en la pantalla la tabla de multiplicar de un número cualquiera, pero sin límite. El organigrama quedaría como en la figura 1. Como BASE se dará entrada al número cuya tabla de multiplicar se desea visualizar. El campo NÚMERO corresponderá a los diferentes valores por los cuales, en incrementos de uno, la BASE irá multiplicándose, dando así un resultado que se visualiza, y vuelta a empezar. No existirá un final hasta llegar al límite de capacidad numérica del ordenador o bien hasta que la persona que observa el desarrollo del programa adopte una decisión.

Los *bucles condicionados* se caracterizan porque su final depende siempre de una decisión, una pregunta que sirve de test para enviar la secuencia de nuevo a ejecutar la repetición o bien salir del ciclo. Retomando el ejemplo de la tabla de multiplicar, supongamos que se quieren visualizar los diez primeros elementos de la tabla. El programa sería el que indica la figura 2. La novedad estriba en la inclusión del rombo que marca una decisión. La pregunta sobre si el número, convenientemente incrementado tras el cálculo e impresión del resultado, ha superado el valor 10 tendrá dos posibles salidas. Si no se cumple dicha condición, es decir, si el número todavía está dentro de los 10 primeros números, el flujo retorna al principio del ciclo, mientras que en el caso contrario la secuencia derivará hacia el final del organigrama.

Es posible en cualquier momento aumentar o disminuir el intervalo, o sea el número de veces que el ciclo va a ser repetido, solamente con variar el valor incluido en el símbolo decisorio.



Un mismo programa puede contener un número no determinado de bucles, inconexos o no entre sí. En el próximo capítulo estudiaremos los denominados bucles anidados.

Tabla de multiplicar hasta el infinito

INICIO
ENTRAR BASE
NÚMERO = 1
RESULTADO = BASE × NÚMERO
RESULTADO
NÚMERO = NÚMERO + 1

Figura 1

Tabla de multiplicar hasta 10

INICIO
ENTRAR BASE
NÚMERO = 1
RESULTADO = BASE × NÚMERO
RESULTADO
NÚMERO = NÚMERO + 1
NÚMERO > 10
NO
SÍ
FIN

Figura 2

La evolución del Dragon

El Dragon 64 es una versión mejorada del 32, que ofrece amplias posibilidades para un serio software de gestión

El Dragon 64 posee el mismo teclado que los últimos Dragon 32, si bien es de mejor calidad. Pero carece de muchas teclas útiles, como Escape, Tab y Control.

Al encenderlo por primera vez, el Dragon 64 (al igual que el 32) estará automáticamente en la "modalidad 32", en la que puede ejecutar todo el software de Dragon. Los usuarios de BASIC disponen de hasta 30 Kbytes de memoria libre. La orden EXEC 48000 apaga la ROM de BASIC y enciende en la parte superior 32 Kbytes de memoria libre. El BASIC se copia entonces en la memoria de arriba, dejando hasta 45 Kbytes libres. Los 64 Kbytes completos sólo están disponibles bajo los otros sistemas operativos o programas en código de lenguaje máquina. El DOS de Dragon restaurará el sistema a la modalidad 32, dejando sólo alrededor de 23 Kbytes libres para los programas en BASIC.

Sorprendentemente, el tablero de circuitos impresos del 64 es muy diferente del tablero del 32. Se trata del mismo ordenador, pero se han desplazado los chips con el objeto de dejar sitio a la nueva puerta en serie y la memoria extra. A consecuencia de ello, los usuarios del 32 que deseen mejorar sus máquinas habrán de reemplazarlas por el modelo 64, ya que no podrán ampliar sus modelos 32 enchufando componentes extras. La CPU del 64 es la Motorola 6809, un diseño de ocho bits que ha llegado demasiado tarde para alcanzar la popularidad de la 6502 y la Z80. Un chip compañero, el 6847, genera la visualización ya sea en un aparato de televisión o en un monitor de video compuesto. Es este chip el que le confiere al Dragon sus más bien curiosas modalidades de visualización.

El Dragon cuenta con una pantalla para textos de 32 × 16 caracteres sobre un fondo verde o naranja. Esta visualización limitada, si bien no constituye un problema para el usuario del Dragon 32, tiene serias implicaciones para quienes deseen utilizar el Dragon 64 como máquina de oficina. El nuevo software basado en disco, en particular el sistema operativo OS9 profesional, se ve muy limitado por la reducida superficie de visualización.

Para gráficos detallados el Dragon posee modalidades en alta resolución, que van desde una resolución de 128 × 96 en cuatro colores hasta un máximo de 256 × 192 en dos colores. En comparación con muchas otras máquinas, esto es muy limitado, especialmente porque la calidad de la visualización puede ser pobre. Sin embargo, le permite al Dra-

Modulador
El modulador proporciona una señal de imagen y sonido para un aparato de televisión

Mando reajuste (reset)
Este interruptor de botón restaura el ordenador sin que se pierda el programa de la memoria

Interfaces
Las puertas para cassette, palanca de mando y en serie están situadas en el lado izquierdo

Chip de E/S
Dos chips 6821 se ocupan de las puertas de entrada y salida del Dragon

Conexión impresora
Una puerta Centronics estándar permite conectar la mayoría de impresoras

CPU
El Dragon está basado en el microprocesador Motorola 6809. Este chip es muy potente y fácil de programar

Conector terminal
Se utiliza tanto para el software en cartucho como para ampliación, igual que la interface de disco Dragon

RAM
En el tablero existe una RAM completa de 64 K, a pesar de que sólo 45 K están disponibles para los programadores en BASIC

Chris Stevens





La unidad de disco Dragon
La unidad de disco Dragon posee una o dos unidades de 175 K cada una. Viene con el DOS de Dragon, pequeño pero adecuado, incorporado en su cartucho de interface



Enchufe red

Controlador de visualización
El chip 6847 genera la visualización para televisor y monitor del Dragon

Disipador
El transformador de potencia del Dragon está en una caja separada, pero aun así el sistema necesita un gran disipador para conservarse frío

BASIC
Estas dos EPROM retienen el BASIC Microsoft de 16 K del Dragon

El Dragon 32
El ordenador personal 32 aún está a la venta y es muy popular. Pero muchos propietarios están mejorando sus máquinas, para equipararlas con el poder y la flexibilidad del 64



Ian McKinnell

gon ejecutar juegos de estilo recreativo y se puede utilizar con el software escrito especialmente para visualizar mayúsculas y minúsculas en 51 columnas de 24 filas, útil para programas de hoja electrónica y de tratamiento de textos.

Almacenar y ampliar

El Dragon posee una buena versión de BASIC Microsoft. Es una versión ampliada de 16 Kbytes y entre sus facilidades figura un conjunto de eficaces órdenes para gráficos y sonido. El Dragon puede retener simultáneamente en la memoria un gran número de imágenes de gráficos en pantalla y puede conmutar entre ellas de forma instantánea, proporcionando una forma sencilla de realizar animación de gráficos en BASIC. El usuario también puede optar por emplear el espacio de memoria normalmente reservado para gráficos para los programas y los datos en BASIC.

La máquina posee un conjunto completo de interfaces, incluyendo dos conversores de analógico a digital y una puerta Centronics estándar para conectar la mayor parte de impresoras en paralelo. La nueva puerta en serie puede conectar a impresoras (incluyendo impresoras margarita) o a otros ordenadores y equipos. Una forma interesante de desarrollar el sistema sería conectarlo a un terminal profesional, lo que permitiría la utilización de un software más avanzado.

El Dragon se conecta en interface a grabadoras de cassette estándar y puede hacer arrancar y detener la cinta desde dentro del programa, y reproducir sonido desde la cinta a través del altavoz del televisor. El BASIC también admite una gama de órdenes para manipulación de archivos en cassette. Ya hemos analizado las unidades de disco del Dragon (véase p. 584). Vienen con un efectivo sistema operativo en ROM en el cartucho de la interface, y con una serie de ampliaciones al BASIC.

El Dragon 64 no se podrá ampliar para convertirlo en un serio sistema de gestión hasta que no posea una adecuada visualización en pantalla y un teclado profesional. Ahora se trata de una máquina muy capaz e interesante para aficionados.

El OS9 multitarea-multiusuario
Su microprocesador 6809, sus 64 Kbytes de memoria y sus unidades de disco le permiten al Dragon 64 ejecutar el sistema operativo OS9 profesional. Es éste el principal sistema operativo para ordenadores basados en el 6809, y proporciona algunas facilidades excelentes así como el acceso a una gama de sofisticados paquetes de gestión. Dragon ha dedicado considerables esfuerzos a lograr que el 64 dispusiera del OS9. Pero el elevado costo del OS9 y de su software podría limitar su popularidad entre los usuarios de ordenadores personales.
El OS9 proporciona las facilidades estándar asociadas con el UNIX, un sistema operativo para miniordenadores de que disponen los micros de gestión más grandes. Entre sus configuraciones se incluye la capacidad para que un usuario ejecute más de un programa a la vez (multitarea) y para que más de una persona utilice el ordenador al mismo tiempo (multiusuario). Para conseguir estas capacidades, el OS9 organiza los archivos de programas y de información en una estructura formal, en vez de limitarse a hacerlo en un directorio simple de archivos en cada disco. Cada archivo posee también contraseñas y códigos de acceso, de modo tal que a ciertos programas y a cierta información sólo pueden acceder determinadas personas.
La mayoría de los usuarios de ordenadores personales no explotan estas facilidades y el Dragon debería soportar una tensión considerable si se utilizara el OS9 al completo. No obstante, proporciona una forma económica de experimentar con un sistema sofisticado, además de dar acceso a programas avanzados y a lenguajes como el C y el PASCAL

DRAGON 64

DIMENSIONES
380 × 330 × 90 mm

CPU
6809

MEMORIA
64 K de RAM, de los cuales hasta 45 K están disponibles para programas en BASIC.
16 K de ROM

PANTALLA
En la modalidad para textos, 16 filas de 32 columnas, mayúsculas sólo con un juego de formas para gráficos en ocho colores. Modalidades para gráficos desde 128 × 96 en cuatro colores, hasta 256 × 192 en dos colores

INTERFACES
Palancas de mando (2), puerta en serie, puerta para impresora en paralelo, puerta para cassette, monitor compuesto con sonido, puerta para TV y ampliación con cartucho

UNIDADES DE DISCO
Hasta dos de 175 K, con el DOS de Dragon o bien con el sistema operativo OS9

LENGUAJES DISPONIBLES
BASIC, FORTH, lenguaje ensamblador 6809. El OS9 proporciona C, PASCAL, BASIC estructurado, así como otros lenguajes

TECLADO
Tipo máquina de escribir con 53 teclas

DOCUMENTACION
Lamentablemente, los manuales del Dragon contienen sólo la información más elemental y, por otra parte, adolecen de omisiones y errores

VENTAJAS
El 64 posee las configuraciones de un ordenador sofisticado (una gran memoria y una gama completa de interfaces). El sistema de disco constituye un elemento valioso, y la capacidad de ejecutar software OS9 una propiedad de primer orden, aunque su precio es relativamente alto

DESVENTAJAS
Sufre de limitaciones en cuanto al teclado y las modalidades de visualización en pantalla, especialmente cuando se utiliza para tareas serias. La CPU 6809 de 1 MHz podría resultar demasiado lenta para algunas aplicaciones determinadas, basadas en el OS9

Escaparate de color

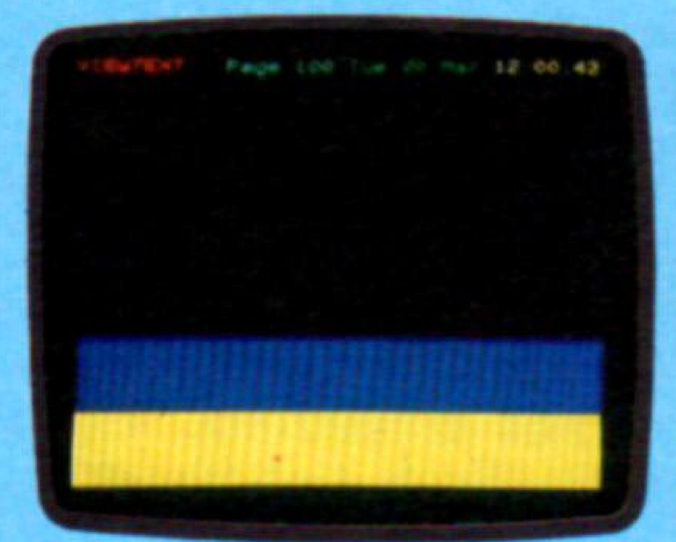
Con un adecuado software para gráficos es posible crear páginas de videotex y hacerlas desfilar cíclicamente por la pantalla

Además del Prestel, en Gran Bretaña existen otros dos sistemas públicos de videotex: Ceefax y Oracle, aunque dado que se transmiten a través de las ondas y que no se envían mediante un cable, se los denomina estrictamente sistemas de *teletexto*. Muchas grandes empresas han adoptado sus propios sistemas de videotex "de uso doméstico", que mantienen al cuadro directivo informado con índices y sucesos de la empresa y les permite acceder a información especializada (como las últimas cotizaciones bursátiles, p. ej.). Siguiendo la misma tendencia, algunos municipios británicos han instalado un sistema de videotex local y gratuito, que contiene la información allí denominada *What's on* (espectáculos y efemérides), detalles relativos a los servicios municipales, etc.

En los centros comerciales urbanos el videotex también se está convirtiendo en algo común. Tal vez un agente de viajes haya hecho las reservas de vacaciones a un cliente mediante el Prestel, o quizá en la tienda de alquiler de películas de video haya un sistema de videotex en el escaparate, que va mostrando una serie de páginas de publicidad, según el sistema que se conoce como carrusel.

En todas estas áreas se puede instalar una fuente informativa. Esto significa que uno puede crear páginas de información en videotex y después utilizarlas en un sistema privado de videotexto propio o venderlas a los clientes. Por ejemplo, en Gran Bretaña colocar una página individual en el sistema Prestel vale entre 15 y 45 libras, pero después de colocar la información se suele cargar a los usuarios del Prestel cada vez que leen dicha página (normalmente entre uno y 10 peniques por cada vez que se accede a ella). A juzgar por la cantidad de fuentes informativas posibles en el Prestel, resulta evidente

Creación de páginas
Vamos a seguir uno a uno los pasos necesarios para crear una página de videotex. Supongamos que una agencia de viajes local le ha encargado que cree un llamativo anuncio ofreciendo unas vacaciones en Grecia, el cual, junto a otras páginas, se colocará en un sistema carrusel en el escaparate de la agencia. El primer paso, antes de que toquemos siquiera el teclado, consiste en dibujar en un papel un boceto esquemático de la página. Para ser más exactos, podríamos trazar primero una cuadrícula de 40 por 24 y luego trasladar cada cuadrado del papel a un carácter de pantalla. Dibujaríamos una escena típica de vacaciones (es decir, playa, mar, sol...) y a continuación superpondríamos los detalles de precio, lugar, etc.



Página 100
Aquí hemos dibujado el mar y la playa. El mar se produce pulsando la tecla de función azul seguida por la tecla de fondo al comienzo de cada línea. Varias líneas de fondo amarillo componen la playa



Página 101
Ahora empezamos a "construir" la colina olímpica con caracteres mosaico verdes superpuestos sobre el mar. De forma similar, se han realizado los primeros indicios de una sombrilla con mosaicos color magenta



Página 102
La imagen toma forma: se ha terminado la colina, a la sombrilla se le han agregado franjas rojas y se ha comenzado a esbozar el sol. Cada una de las letras grandes se compone de seis caracteres mosaico



Página 103
La imagen ya está casi acabada. Las letras grandes, el sol y la sombrilla ya están completos y se ha agregado un barco de vela. Para dar un toque de autenticidad, ¡no falta ni siquiera el templo griego mirando al mar!



Página 104
La página al completo. El toque final consistió en la adición del texto digitado sobre la imagen. La leyenda "TWO WEEKS IN GREECE" (dos semanas en Grecia) se hizo con caracteres de doble altura, y si no se tratara de una foto fija podríamos ver que en realidad el precio es intermitente. Es de advertir que las "olas" que hemos agregado al mar son signos menos (−) de color cyan

Ian McKinnell

que se puede ganar mucho dinero, ya sea de manera directa a través de ese cargo por acceder a la página o bien indirectamente a través de la publicidad conseguida a partir de la misma.

Un emprendedor grupo de estudiantes de Gales del Sur se pusieron en contacto con unos grandes almacenes durante la época de Navidad y se ofrecieron para instalar un sistema de videotex local que visualizara para los clientes páginas de anuncios. Crearon las páginas utilizando un sencillo paquete editor de videotex con su micro y después instalaron el ordenador en el escaparate principal de la tienda, donde automáticamente iban pasando una y otra vez las diversas páginas. ¡Cobrando de la tienda las páginas y el alquiler del micro se ganaron una buena paga extra de Navidad!

Éstos son apenas algunos ejemplos de las aplicaciones que puede tener el videotex, pero a cualquiera que tenga un poco de imaginación seguramente se le ocurrirán otros muchos. Todo cuanto se necesita es un microordenador, un poco de talento artístico y, por supuesto, el software.

El paquete de videotex específico que vamos a analizar aquí se denomina *Viewtext* y funciona en el BBC Micro, ordenador que se destaca sobre la mayoría de los otros micros en virtud de su juego de caracteres de videotex incorporado. Aun siendo este paquete uno de los más baratos, incorpora tanto un editor de videotex para crear las páginas como un programa para establecer un sistema carrusel de 22 páginas.

El editor trabaja de forma muy similar a cualquier otro sistema de edición basado en pantalla: el cursor se posiciona mediante las teclas de control del cursor y después se digita el texto. Para seleccionar las diversas características del videotex (colores para el texto y para el fondo, caracteres intermitentes, caracteres de altura doble, etc.), tan sólo debe pulsar las teclas de función rojas de la fila superior del teclado del BBC. Por ejemplo, para conseguir que la palabra "HOLA" aparezca intermitente y en letras amarillas, se pulsan las teclas f3 (para amarillo) y f8 (para intermitente), y después se digita la palabra.

Habiendo llegado a este punto, es importante mencionar dos detalles respecto al funcionamiento de un editor de videotex. En primer lugar, las propiedades como "intermitente" o "texto rojo" se pierden cuando se pasa a la línea siguiente: el texto vuelve a las letras blancas y pequeñas sobre fondo negro. En segundo lugar, cada característica ocupa un espacio de carácter invisible en la pantalla. Por lo tanto, en el ejemplo anterior habría dos espacios aparentemente en blanco delante de la palabra "HOLA". Si usted borra estos espacios o si escribe sobre ellos, entonces las propiedades correspondientes se pierden.

Gráficos de videotex

Lo que hemos dicho hasta ahora nos permite crear páginas que sólo contengan texto, lo cual resulta adecuado para muchas aplicaciones de videotex. Sin embargo, los diagramas, el material publicitario y cosas de este tipo en realidad exigen no sólo la utilización de texto sino también de gráficos. Los gráficos de videotex se crean a partir de un juego de caracteres para gráficos, constando cada uno de los caracteres hasta el seis pixels en una cuadrícula de dos columnas por tres filas. Las imágenes se construyen de forma similar a como se construye un mosaico mediante ínfimas teselas y por esta razón a los caracteres se los denomina *mosaicos alfa*. La calidad de la imagen resultante deja bastante que desear si se compara con las capacidades para gráficos de la mayoría de los ordenadores personales, pero con un poco de imaginación cualquier imagen se puede realizar con mosaicos alfa.

A este propósito, añadiremos que la razón por la cual los gráficos son tan elementales es que a cada página de videotex sólo se le concede un Kbyte de memoria, mientras que una pantalla para gráficos de alta resolución de un ordenador personal puede utilizar alrededor de 20 Kbytes. Dada la velocidad baudio a la que funciona el Prestel, el tiempo que se tardaría en recibir una página completa en alta resolución sería excesivo.

La utilización de los caracteres mosaicos alfa es un área en la que el editor *Viewtext* no se desenvuelve bien. Los caracteres se obtienen pulsando primero la tecla de función roja (f9) para encender los gráficos, y pulsando después cualquiera de las teclas de minúsculas o de números del teclado. Cada tecla produce un carácter para gráficos, pero lamentablemente no existe relación alguna entre lo que está grabado en cada tecla y el carácter que aparece en la pantalla: uno debe remitirse continuamente a una tabla del manual para encontrar la tecla adecuada. Este proceso es muy pesado: la creación de los gráficos mosaico en las fotografías que acompañan este artículo requirió un par de horas. Los editores más sofisticados permiten que uno construya cada carácter encendiendo o apagando cada uno de los seis pixels de la matriz del carácter, método éste mucho más rápido.

Hay dos configuraciones especiales que sólo se utilizan con los caracteres para gráficos de mosaico: *Separate graphics* (separar gráficos) hace saltar cada carácter de modo que los pixels queden ligeramente separados los unos de los otros; *Hold graphics* (retener gráficos) se utiliza para cubrir los espacios en blanco que quedan cuando se establece una propiedad nueva. Por ejemplo, para obtener un carácter de mosaico rojo junto a otro verde sin que quede un espacio en blanco entre ellos, se debe pulsar la tecla de función *Hold graphics* antes de pulsar la tecla *Green graphics* (gráficos verdes).

Con esto cubrimos la mayoría de las configuraciones de que disponen todos los editores de videotex; las versiones más caras poseen facilidades para acelerar el proceso de creación de páginas, que re-

El editor Viewtext
El paquete Viewtext incorpora un sencillo editor para que el usuario pueda diseñar sus propias páginas de teletexto. En este ejemplo, el anuncio de las vacaciones en Grecia se ha adaptado modificando el texto y la imagen

Videotex de alta resolución

Ian McKinnell

La mayoría de los sistemas de videotex utilizan gráficos de mosaicos alfa, que son económicos en cuanto a memoria pero menos satisfactorios en cuanto a la calidad de las imágenes obtenidas. El estándar alternativo para gráficos con videotex se denomina alfageometría y es capaz de producir gráficos mucho más realistas.
Además de las letras, los números y los caracteres mosaico usuales, los gráficos alfageométricos permiten dibujar líneas, círculos y formas similares utilizando órdenes simples, de forma muy parecida a las órdenes en BASIC para gráficos de alta resolución de que disponen la mayoría de los micros personales. Para acomodar la resolución más alta, la pantalla se divide en 320 × 240 pixels, frente a los 40 × 24 cuadrados que se utilizan en los gráficos de mosaicos alfa. Esto tiene el inconveniente de que transmitir a través de las líneas telefónicas una imagen muy detallada (p. ej., la del rostro de una persona) ocupa un par de minutos.
Uno de los sistemas comerciales disponibles en el mercado británico es el terminal MUPID. Se trata de un microordenador exclusivo y es capaz de visualizar tanto gráficos de mosaicos alfa como alfageométricos. Entre sus configuraciones más novedosas está la capacidad de producir animación en una página individual, así como colores de tonalidad intermedia y sombreado.
El sistema alfageométrico es excelente para presentar imágenes detalladas, como mapas o facsímiles.

sultan útiles en el campo comercial pero que no son necesarias para el usuario personal. Por ejemplo, un paquete más caro le permitiría desplazar o copiar partes de la página de videotex y contaría con un método más rápido para crear los gráficos de mosaico que los descritos anteriormente.

Después de que haya creado su página de videotex, el paso siguiente consiste en guardarla en disco o en cinta, otorgarle un número de página y colocarla en su propio sistema de videotex. El paquete *Viewtext* puede retener hasta 22 páginas en la memoria del BBC en cualquier momento, y a cada página se le da un número entre 100 y 121. Luego que haya creado y guardado suficientes páginas, puede ejecutar el programa carrusel. Éste carga todas las páginas y las hace desfilar en un ciclo repetitivo, siendo el usuario quien decide la duración del intervalo entre página y página. En la línea superior de todas las páginas de videotex aparece información como la fecha y la hora, el número de la página que se está contemplando y el nombre del servicio de videotex (Prestel, etc.). El programa carrusel pide al usuario esta información antes de ir pasando las páginas cíclicamente.

Hay paquetes de videotex más refinados que permiten transmitir y recibir páginas de otros sistemas, utilizando un modem y una línea telefónica. Estos paquetes son caros y exigen gran cantidad de almacenamiento en disco para las muchas páginas de información, de modo que podrían no estar al alcance de la mayoría de los usuarios de ordenadores personales. Ejecutando un sencillo programa para edición en un micro personal, se encuentra usted en disposición de crear páginas de videotex capaces de competir con las que se obtienen de cualquier agencia comercial.



Espacio de memoria

Debemos analizar dónde van a parar los programas dentro de la memoria, y ver cómo se lleva a cabo una operación aritmética simple

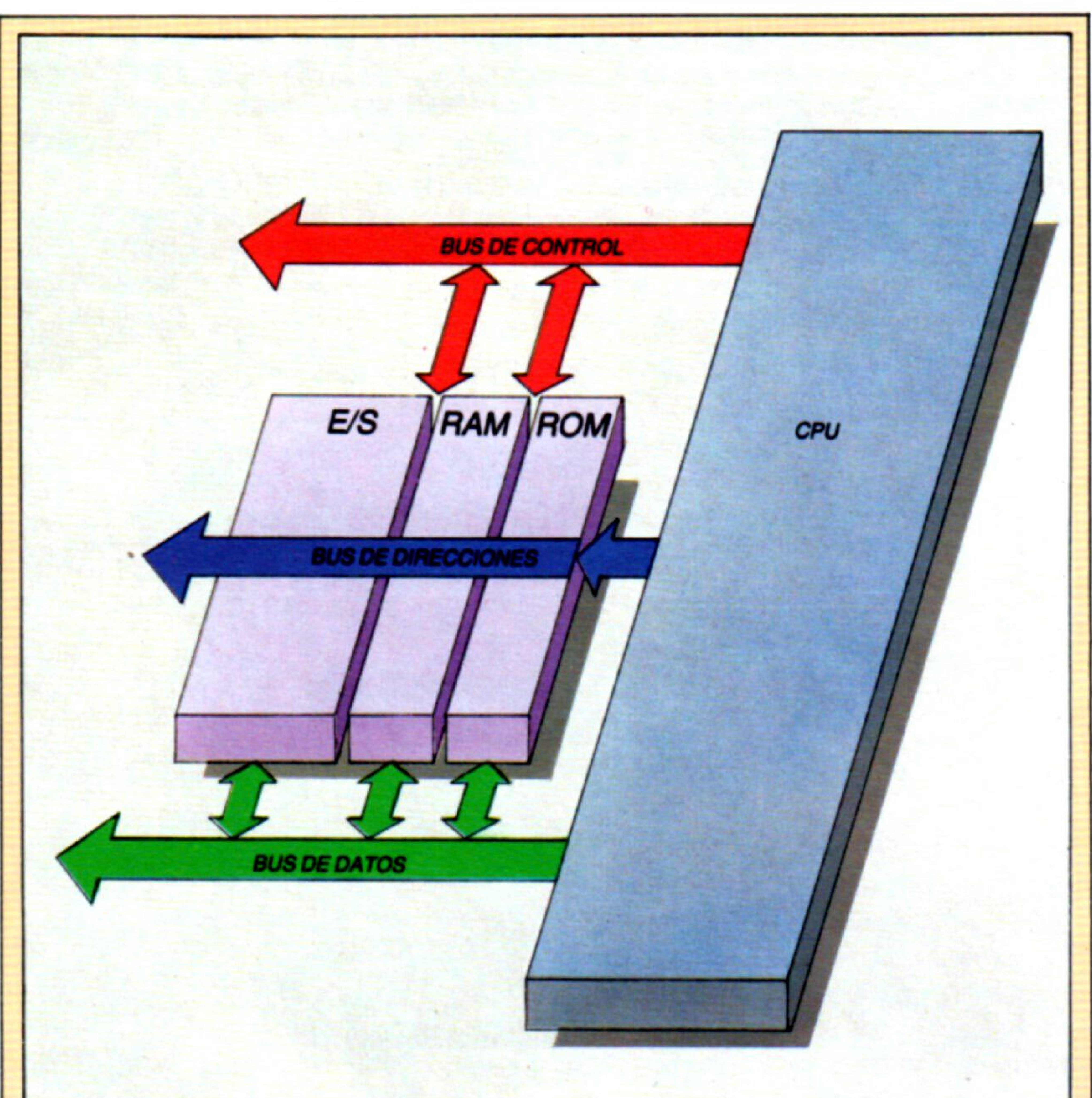
En la última lección de este curso de lenguaje máquina elaboramos un programa muy sencillo en lenguaje ensamblador, lo tradujimos (ensamblamos) a lenguaje máquina, lo cargamos en la memoria y lo ejecutamos. Para las dos últimas tareas utilizamos el programa monitor de la página 598. Si el programa fuera un paquete más sofisticado que contuviera un ensamblador, podríamos haberlo empleado también para ensamblar nuestro programa en lenguaje máquina. En esta etapa del curso, hacer el ensamblaje a mano no supone ningún esfuerzo excesivo; en realidad, constituye un ejercicio muy educativo. Pero una vez que usted haya asimilado los principios del proceso, y a medida que sus programas en lenguaje ensamblador se vayan haciendo más largos, no tendrá ningún sentido que se preocupe por la verdadera traducción a lenguaje máquina. Por consiguiente, cuando llegue a esta etapa del aprendizaje del lenguaje máquina le interesará adquirir un programa ensamblador adecuado para su ordenador.

Son muchos los puntos a resaltar dentro del corto programa en lenguaje máquina que le proporcionamos (véase p. 597). Nosotros utilizamos uno de los registros de la CPU para manipular la memoria, tuvimos que decidir en qué lugar de la memoria íbamos a almacenar el lenguaje máquina e hicimos que el microprocesador lo ejecutara. Todos éstos son aspectos de la programación en lenguaje ensamblador que confunden al principiante y deben analizarse con detención. Empecemos por la cuestión de dónde se almacena el lenguaje máquina.

Para la CPU la única diferencia entre un byte de memoria y el siguiente es el que sea de memoria de acceso directo (RAM) o de memoria de lectura solamente (ROM). Los chips de ROM contienen programas y datos del sistema que se deben proteger contra una sobreescritura accidental o deliberada y, por consiguiente, sólo se pueden leer. En ellos no se puede escribir, de modo que no es posible cargar un programa de lenguaje máquina en la ROM. A excepción de estas áreas de la memoria, teóricamente nada hay que nos impida cargar un programa en cualquier otra parte de la misma, pero existen ciertas consideraciones prácticas que nos prohíben utilizar determinadas áreas.

La CPU emplea ciertas secciones de la RAM para almacenamiento temporal en el curso de sus operaciones, y, si cargamos un programa allí, simplemente se destruirá porque la CPU escribe sobre ella o bien (y esto es lo más probable) la CPU leerá nuestro lenguaje máquina como si se tratara de algunos de sus propios datos. El sistema operativo también utiliza grandes porciones de la RAM para almacenar sus datos de trabajo y para ejecutar el sistema del ordenador. Cargar programas en código de lenguaje máquina en cualquiera de estas zonas sería imprudente o imposible, por las mismas razones que nos prohíben emplear el espacio de trabajo de la CPU. Además, los programas en BASIC pueden ocupar toda la RAM restante (en parte como texto de programas y en parte como zonas de almacenamiento de variables). Una vez más, sería un error tocar estas zonas y por ello el programador se enfrenta con un complejo dilema.

Una respuesta es no utilizar el BASIC para nada, liberando zonas como el área para textos de programas en BASIC. Pero por lo general es convenien-



La arquitectura de un pequeño sistema

Un sistema típico de ordenador, en su forma más esquemática, comprende la memoria y una CPU. La primera se compone de chips de ROM (que contienen programas del sistema), chips de RAM y chips especializados exclusivos para operaciones de entrada-salida. Los datos y las señales de control fluyen hacia y desde la CPU y alrededor del sistema mediante *buses*. Los buses son vías (muy similares a cables planos) que pueden transportar un byte o más de datos por vez. Pueden ser unidireccionales, como el *bus de datos*, que puede transmitir en cualquier sentido. El *bus de control* transporta información de conmutación alrededor del sistema, abriendo y cerrando puertas lógicas para dirigir el flujo de datos. El *bus de direcciones* transporta una dirección de 16 bits desde la CPU para seleccionar un byte de memoria, permitiendo que los datos fluyan a lo largo del *bus de datos*, de ocho bits, hacia o desde el byte

te escribir al menos parte de un programa en ese lenguaje, utilizando subrutinas en lenguaje máquina sólo allí donde el BASIC sea demasiado lento, como por ejemplo, en las visualizaciones animadas sobre pantalla. Si los programas en BASIC y en lenguaje máquina han de coexistir en la RAM, entonces debemos reservar algo de espacio y destinarlo para lenguaje máquina. Podemos desplazar las fronteras del área para textos de programas en BASIC, o podemos encontrar secciones de la misma o de la RAM del sistema operativo que estén temporalmente sin utilizar.

Desplazar las fronteras del BASIC es muy fácil en el BBC Micro, dado que estas direcciones están retenidas en las variables del sistema PAGE, TOP, LOMEM y HIMEM. PAGE, por ejemplo, señala el comienzo del área para textos de programas en BASIC, que suele estar en la dirección $1200. Si ejecutáramos la instrucción:

```
PAGE = PAGE + 500
```

entonces el sistema operativo almacenará los programas en BASIC 500 bytes más arriba de la memoria, dejando libre una zona de 500 bytes para nuestros programas en lenguaje máquina. En otros ordenadores se consigue el mismo efecto colocando (POKE) direcciones más altas en los señaladores del sistema (véanse los mapas de memoria de p. 538). Se puede también reservar espacio bajando la dirección del extremo superior del área para textos de programas en BASIC. En el Spectrum, la orden CLEAR seguida de una dirección realiza exactamente eso. La única restricción en cuanto a la cantidad de memoria que se puede reservar mediante este método es que quede espacio suficiente para nuestro programa en BASIC.

En la RAM del sistema operativo se pueden encontrar pequeños bloques de espacio libre; un ejemplo típico de ello es el buffer (almacenamiento intermedio) para cassette del Commodore 64. Éste se compone de 192 bytes de RAM (desde $033C hasta $03FB) y el sistema operativo sólo lo utiliza cuando se emplea la unidad de cassette. A muchos programadores este espacio les basta para satisfacer todas sus necesidades de lenguaje máquina.

Dentro de programas en BASIC se pueden encon-

La organización interna de la CPU

El Z80 posee una estructura más compleja que el 6502. Los registros se usan como RAM de espacio de trabajo en las operaciones de la CPU, o para tareas específicas como controlar la pila, retener la dirección de la siguiente instrucción del programa, mostrar los efectos de la última operación de la CPU o retener direcciones.
Las dos CPU poseen un acumulador de ocho bits conectado a la ALU, donde se realizan las operaciones aritméticas y lógicas, pero el Z80 también admite algo de aritmética de 16 bits en sus registros pareados (llamados BC, DE y HL)

Kevin Jones



Acumulador
Éste es realmente el registro central de la CPU. Las operaciones de aritmética y lógica, así como las transferencias de datos, se conducen fundamentalmente a través de este registro, con mayor intensidad en el 6502 que en el Z80

Unidad aritmético lógica (ALU)
Consta de un sumador binario y puertas lógicas, que permiten acceder a bits individuales de los registros y al bus de datos. Controlada adecuadamente, posibilita la suma, la resta y las operaciones booleanas

Registro del estado del procesador
Siempre que se realiza una operación de la CPU, los bits individuales del registro del estado del procesador muestran algunos de los efectos de la operación: ¿produce, por ejemplo, un resultado cero?, ¿o hay allí un bit acumulado de la operación suma?

Contador de programa
Señala la dirección de la memoria donde está almacenado el siguiente opcode que va a ejecutar la CPU. La función en BASIC USR(dirección) hace que la dirección especificada se cargue directamente en el contador de programa, de modo que la ejecución de la CPU prosigue a partir de este punto

Indicador de pila
Lleva la dirección del siguiente byte de RAM de espacio de trabajo libre para el empleo de la CPU. Cada vez que la CPU escribe algún dato en él, el indicador de pila cambia para señalar el siguiente byte libre

Registros de índice y uso general
La CPU utiliza los registros de índice para direccionar la memoria de muchas maneras, mientras que los registros de uso general se utilizan como espacio de trabajo general y para fines específicos de la CPU

trar bloques de espacio aún más pequeños: las líneas REM, por ejemplo. En el área para textos de programas en BASIC, esta línea:

```
10 REM*************************
```

tiene 25 bytes consecutivos con el mismo contenido $2A, el código ASCII para '*' (decimal, 42). Ni el sistema operativo ni el intérprete de BASIC inspeccionan jamás estos bytes, porque para ellos la orden REM significa "prescinda del resto de esta línea". Una vez que se le ha dado entrada a la línea en el programa de esta forma, se puede cargar una subrutina en lenguaje máquina en los bytes de asteriscos, donde residirá sin que el intérprete la moleste en lo más mínimo. La gran ventaja que ofrece este método rebuscado en apariencia (pero que con frecuencia se utiliza en programas para el ZX81 ampliado) es que cuando el usuario guarda (SAVE) y subsiguientemente carga (LOAD) el programa en BASIC, la subrutina en lenguaje máquina lo acompaña. La utilización de los otros métodos descritos por lo general significa tener que guardar el lenguaje máquina por separado respecto del programa en BASIC. El problema de este método es que, al LISTar las líneas, el sistema operativo interpreta el byte del lenguaje máquina como datos de caracteres ASCII, lo que produce extrañas visualizaciones en la pantalla. Esto explica por qué había advertencias incluidas en los programas de demostración de la página 499 para el BBC Micro y el Spectrum. La versión de aquel programa para el Commodore 64 carga la subrutina en lenguaje máquina en el buffer de la cassette, mientras que las versiones para el BBC y el Spectrum la cargan en su línea REM de comienzo, y de allí las advertencias de no LISTar estas versiones del programa.

Después de haber escrito un programa en lenguaje ensamblador, de haberlo traducido a lenguaje máquina y de cargarlo (LOAD) en la RAM de su elección, puede proceder a ejecutarlo. Esto se efectúa en BASIC mediante las órdenes CALL (sólo el BBC), SYS (sólo el Commodore 64) o USR (las tres máquinas). Cada una de estas órdenes va seguida de la dirección del primer byte del programa en lenguaje máquina, donde sea que se almacene. Estas tres órdenes significan lo mismo para el intérprete: "Ejecute el programa en lenguaje máquina que comienza en la dirección dada, y retorne a ejecutar la siguiente instrucción en BASIC cuando se ejecute el opcode RET o RTS". Es como GOSUB en BASIC.

En la última lección escribimos un programa para copiar el contenido de un byte en otro byte, cargando en el acumulador el contenido de una dirección y almacenando luego el contenido del acumulador en la otra dirección. Lo que demuestra el papel central que desempeña la CPU en todo el sistema: los datos y el control deben fluir desde la memoria, siempre a través de la CPU, y regresar por este medio a la memoria. Mientras que en BASIC podemos escribir LET X = Y (que significa "copiar el contenido de Y directamente en X"), en lenguaje ensamblador tenemos que copiar de la memoria a la CPU, y de ésta devolver la copia a la memoria. Los registros de la CPU (véase el cuadro adjunto) son los bytes de RAM, dentro de la propia CPU, donde se almacenan o se manipulan los datos de la memoria. Tanto el Z80 como el 6502 poseen un registro llamado *acumulador*, al que se orientan muchas de las instrucciones en lenguaje ensamblador, y que es el registro en el que se efectúan principalmente las operaciones aritméticas.

Supongamos que deseamos sumar $42 y $07 (el símbolo $ significa hexadecimal). Colocamos uno de ellos en el acumulador y agregamos el otro encima del primero: su suma se "acumulará" en el registro. He aquí las instrucciones:

Z80		6502	
LD	A,$42	LDA	#$42
ADC	A,$07	ADC	#$07

Aquí las dos instrucciones del Z80 proveen para que los números se carguen y sumen, mientras que en la versión para el 6502 los números van precedidos por #, que indica que se trata de números verdaderos y no de direcciones. Así, por ejemplo, LDA #$65 significa "cargar el número $65 en el acumulador", mientras que LDA $65 significa "cargar en el acumulador el contenido del byte cuya dirección es $65". Del mismo modo, la instrucción de suma, ADC (que resulta ser una expresión mnemotécnica válida tanto para el Z80 como para el 6502) significa en este caso: "agregar en el acumulador un número verdadero". Los números $42 y $07 son "datos inmediatos", y LDA #$42 se lee como: "cargar en el acumulador los datos inmediatos #$42".

Después de que se hayan ejecutado estas dos instrucciones, el acumulador contendrá la suma de los números. Para nosotros la suma queda allí "invisible", lo que significa que debemos almacenar el contenido del acumulador en un byte de RAM donde se pueda inspeccionar. El programa debe terminar con una instrucción RETURN y comenzar con una instrucción para colocar un registro relacionado de la CPU en el estado correcto, de modo que el programa completo se lee así:

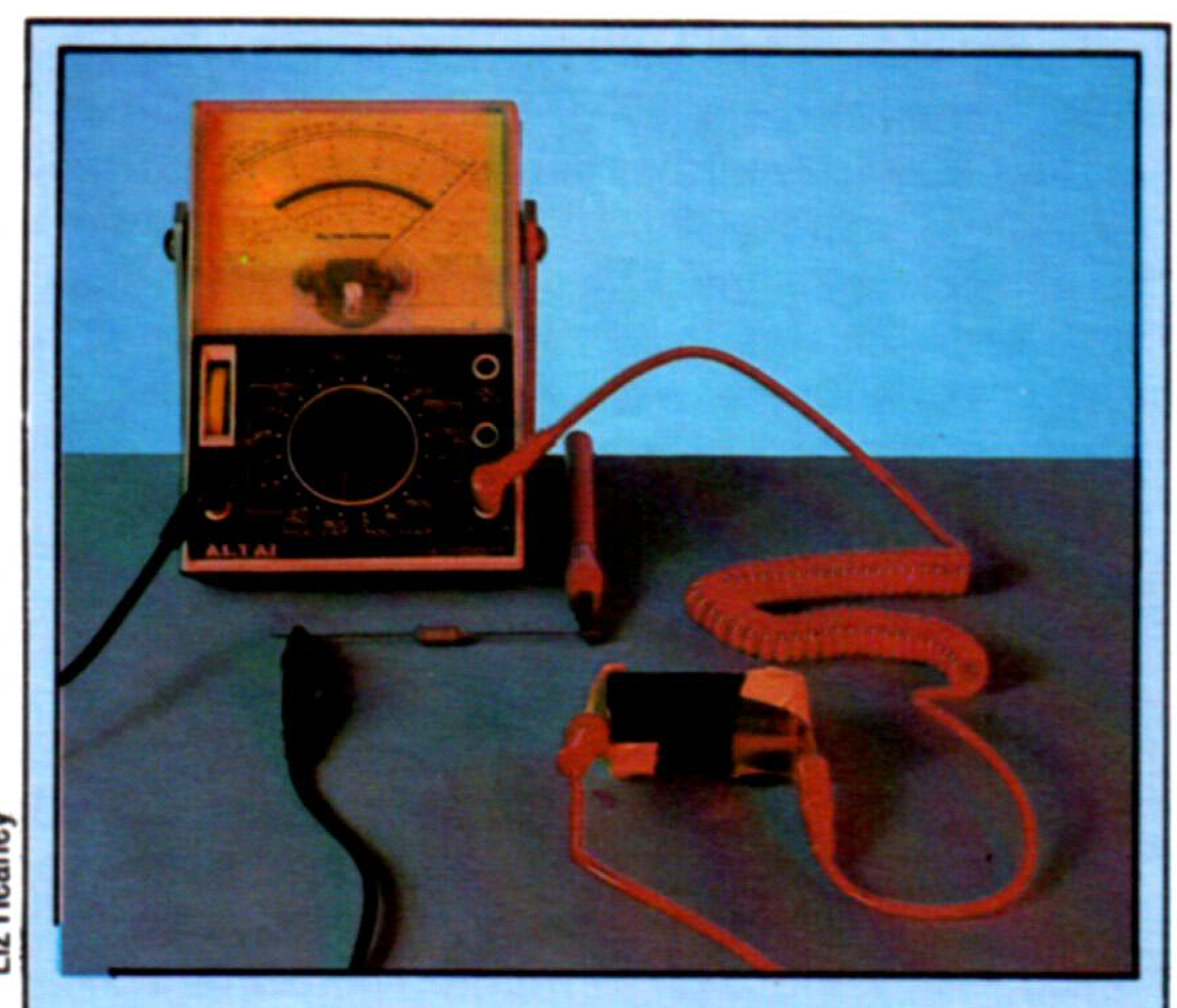
Z80		
Leng. máquina	**Leng. ensamblador**	
A7	AND	A
3E 42	LD	A,$42
CE 07	ADC	A,$07
32 ?? ??	LD	BYTE 1,A
C9	RET	
6502		
18	CLC	
A9 42	LDA	# $42
69 07	ADC	# $07
8D ?? ??	STA	BYTE 1
60	RTS	

De momento no nos vamos a ocupar del significado de la primera instrucción en ambos programas, pero observe que la cuarta instrucción contiene el símbolo BYTE 1en lugar de la dirección verdadera. El valor de BYTE 1 es distinto de una máquina a otra, de modo que aquí nos limitaremos a utilizar tan sólo el símbolo, y más adelante, cuando lleguemos a ensamblar el código, lo sustituiremos por un número hexa real.

Ahora lo que debemos hacer es decidir dónde colocar el lenguaje máquina y qué dirección representa BYTE 1. Elija un lugar para almacenar el lenguaje máquina y después haga que BYTE 1 sea igual a la dirección del byte después del final del programa, y coloque esa dirección en el lenguaje máquina en forma *lo-hi* (bajo-alto). Después de ello, utilice el programa monitor de la página 598 para cargar y ejecutar el lenguaje máquina y para inspeccionar el byte donde se ha de almacenar el resultado: $49.

Partes de una pieza

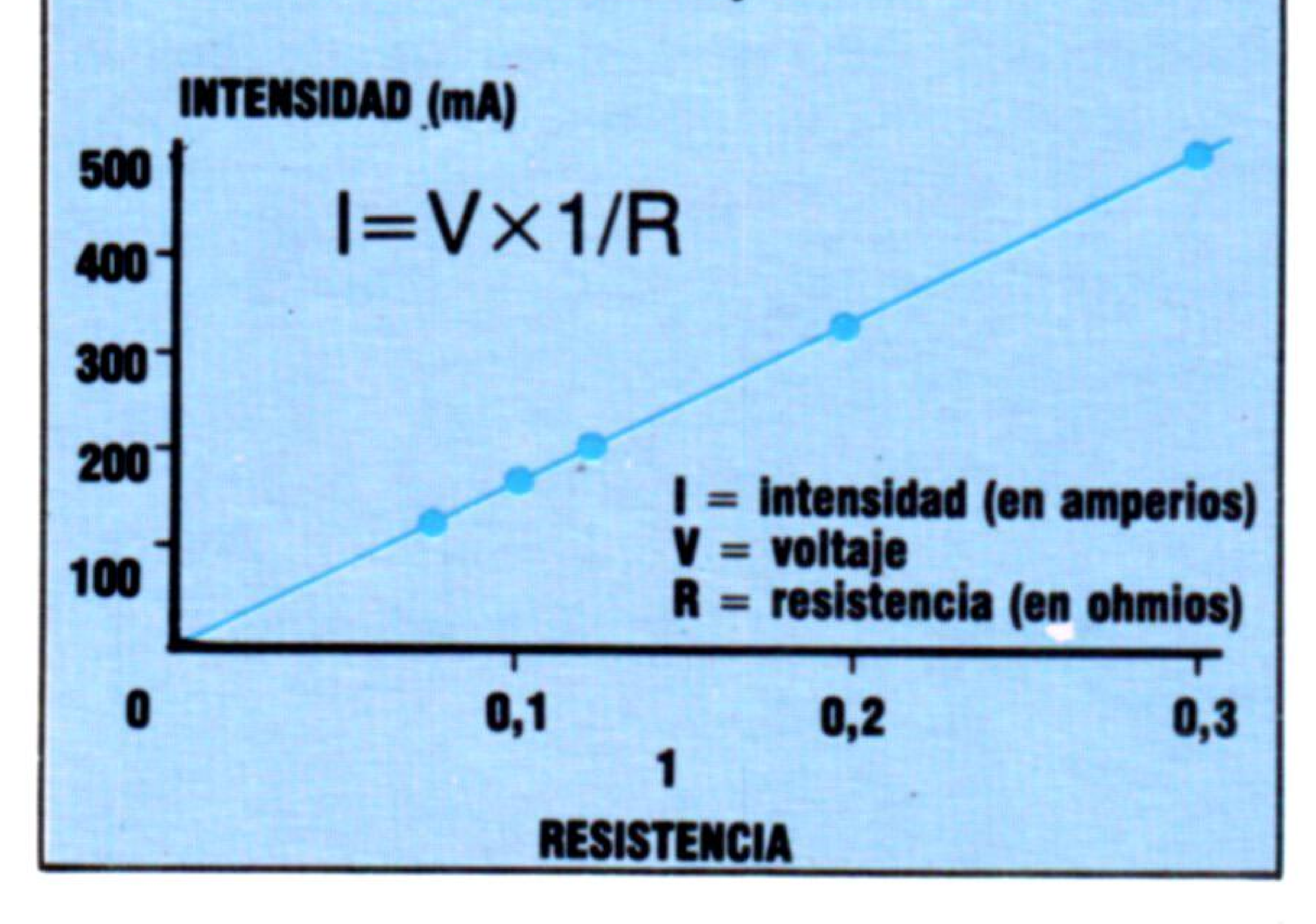
Examinaremos los componentes fundamentales de una pieza y veremos dos modos de calcular los valores de las resistencias



Liz Heaney

Demostrar la ley de Ohm

Se demuestra la ley de Ohm con unas resistencias, una pila, un poco de cable y un multímetro. Las resistencias se valoran a un vatio o más, con valores desde 3,3 ohmios hasta 15 ohmios. Prepare el multímetro para medir la corriente, a la escala de 400 mA (miliamperios) para el indicador. Conéctelo al circuito con los otros componentes, pruebe con cada resistencia y verifique la lectura del multímetro. Si representa gráficamente sus lecturas de corriente en función de uno dividido por los valores de las resistencias tendrá un gráfico de línea recta en el cual I es directamente proporcional a 1/R con un voltaje constante. Luego $I = V \times 1/R$, que es $V = I \times R$, fórmula de la ley de Ohm



Diodos



Los diodos son el equivalente eléctrico de las válvulas unidireccionales. Un diodo posee una resistencia muy baja a la corriente eléctrica en una dirección (por lo general, unos pocos ohmios) y una resistencia muy alta a la corriente en la otra dirección. Esta irregularidad respecto a la ley de Ohm se debe a que el diodo es un semiconductor construido con materiales como el silicio y el germanio. Cada diodo tiene una franja en el extremo hacia el cual puede fluir la corriente

Los microprocesadores y chips de control de video existentes en el corazón de todo microordenador son tan complejos que el amplio alcance de sus acciones y sus diminutas dimensiones hacen que parezcan ajenos al mundo de los humanos.

Aun el más complicado de los circuitos integrados necesita el apoyo de piezas eléctricas mucho más sencillas. Estos componentes simples se pueden clasificar en dos tipos: activos y pasivos.

Los componentes pasivos son los más sencillos. Se limitan a obstaculizar la señal eléctrica que fluye a través de los mismos. Las diferentes señales se obstaculizan de diferentes formas. Por esto resultan tan útiles. Las resistencias y los condensadores constituyen ejemplos de este tipo de componentes.

Los componentes activos tienen más ajetreo. Pueden incrementar la señal que fluye a través de ellos. Un transistor es un buen ejemplo.

Los componentes pasivos se fabrican con materiales simples. Aparte de los materiales de recubrimiento (los diversos plásticos y resinas) que uno ve cuando compra, por ejemplo, una resistencia, el componente está hecho de cobre, acero y carbón. Todos son conductores. Los componentes activos, en cambio, son de silicio o germanio principalmente. Estos dos elementos poseen propiedades especiales que hacen que no se comporten ni como conductores, como el acero y el cobre, ni como aislantes, como las resinas y plásticos. Bajo determinadas condiciones conducen la electricidad, y bajo otras no. Por ello se dice que son semiconductores.

Los componentes activos, debido a que se basan en semiconductores, funcionan de muchas formas interesantes. Uno no puede afirmar que la aplicación de un determinado voltaje a través de un semiconductor dará por resultado que a través del mismo fluya una corriente eléctrica de cierto valor, ni siquiera que fluya alguna corriente en absoluto. Pero, todos los conductores están sometidos a una ley muy sencilla. La ley de Ohm (véase p. 594) predice la forma en que reaccionará cualquier conductor simple ante la señal eléctrica que fluye a través del mismo. Para la informática, el más importante de estos componentes es el transistor, ya que es la base sobre la cual los ordenadores almacenan y procesan la información. En el próximo capítulo utilizaremos transistores para construir algunas de las puertas lógicas simples que hemos analizado en el apartado de *Ciencia informática*.

Spectrum

```
20 DIM B$(3,7)
25 DIM C$(4,7)
30 INPUT "DE ENTRADA AL COLOR DE LAS
   BANDAS"
40 INPUT "BANDA NUMERO 1",B$(1)
50 IF B$(1)="DORADO" OR B$(1)="PLATEADO"
   THEN GO TO 30
60 INPUT "BANDA NUMERO 2",B$(2)
70 INPUT "BANDA NUMERO 3",B$(3)
80 PRINT
90 FOR I=L TO 3
100 RESTORE 180
110 FOR J=0 TO 9
120 READ C$(1),C$(2),C$(3),C$(4)
130 IF C$(1)=B$(I) THEN GO SUB 1000
140 NEXT J
150 NEXT I
160 PRINT "                    OHMIOS"
170 STOP
180 DATA "NEGRO"."0"."0"."0". "MARRON".
    "1","1"."0"."ROJO"."2"."2"."00".
    "NARANJA"."3"."3"."000"
190 DATA "AMARILLO"."4"."4"."0000".
    "VERDE"."5"."5"."00000"."AZUL"."6".
    "6"."000000"
200 DATA "VIOLETA"."7"."7"."0000000".
    "GRIS"."8"."8".".0"."BLANCO"."9".
    "9".".0"
1000 FOR K=1 TO 7
1020 IF C$((I+1),K)<>" " THEN PRINT C$
     ((I+1),K);
1025 NEXT K
1030 RETURN
```

BBC

```
10 REM CODIGOS DE COLOR DE RESISTENCIAS
20 DIM B$(3),C$(3)
30 PRINT "DE ENTRADA A COLOR DE BANDAS:"
40 INPUT "BANDA NUMERO 1",B$(1)
50 IF B$(1)="DORADO" OR B$(1)="PLATEADO"
   THEN GOTO 30
60 INPUT "BANDA NUMERO 2",B$(2)
70 INPUT "BANDA NUMERO 3",B$(3)
80 PRINT
90 FOR I=1 TO 3
100 RESTORE
110 FOR J= 0 TO 9
120         READ C$(0),C$(1),C$(2),C$(3)
130         IF C$(0)=B$(1)
    THEN PRINT C$(I);
140         NEXT J
150 NEXT I
160 PRINT "              OHMIOS"
170 END
180 DATA NEGRO,0,0,, MARRON,1,1,0,
    ROJO,2,2,00, NARANJA,3,3,000
190 DATA AMARILLO,4,4,0000,
    VERDE,5,5,00000, AZUL,6,6,000000
200 DATA VIOLETA,7,7,0000000,
    GRIS,8,8,, BLANCO,9,9,,
```

Para el Commodore 64 reemplace las líneas 40, 60 y 70 por:

```
40 INPUT "BANDA NUMERO 1";B$(1)
60 INPUT "BANDA NUMERO 2";B$(2)
70 INPUT "BANDA NUMERO 3";B$(3)
```



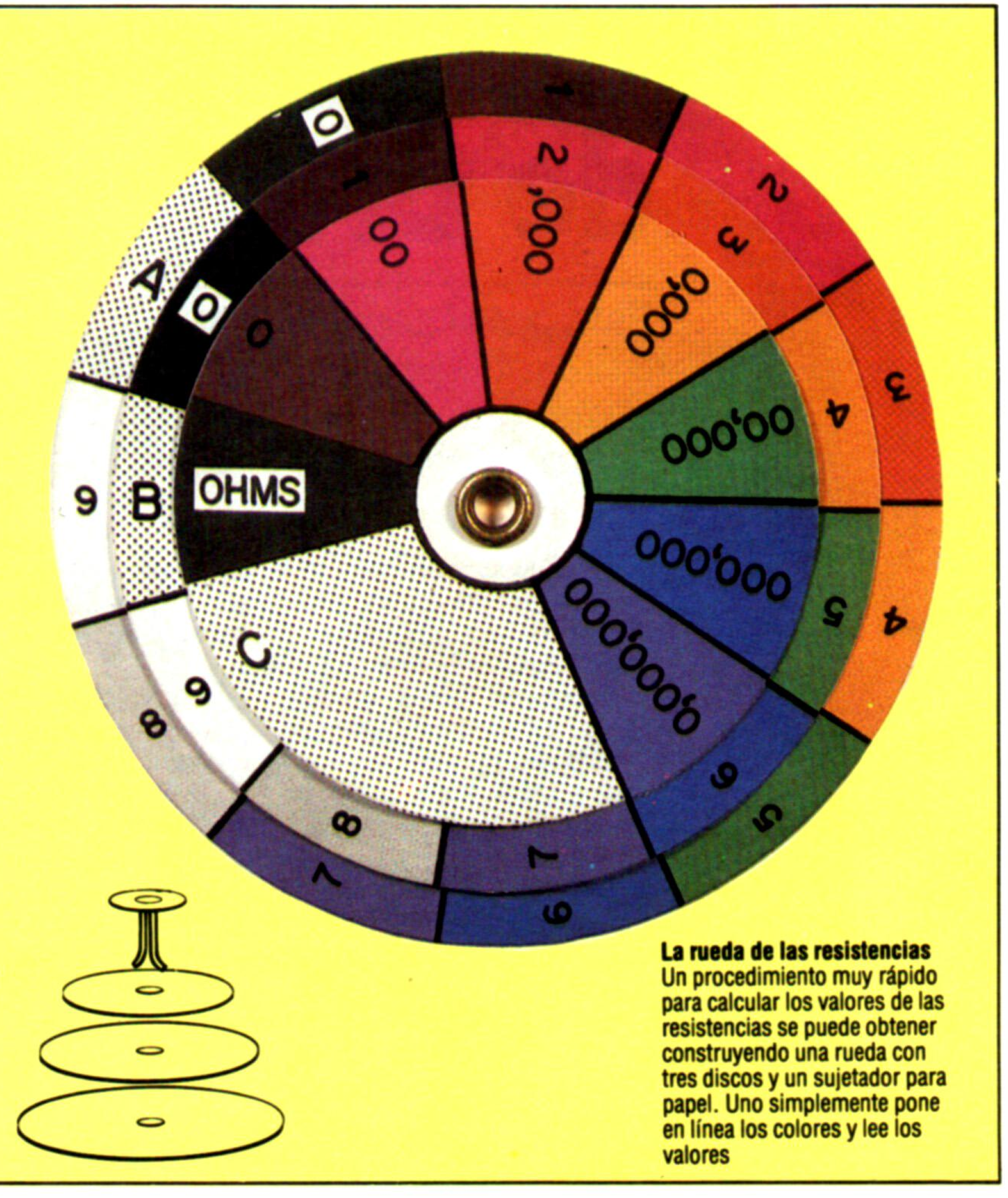
Resistencias

DE ALAMBRE ENROLLADO

DE PELÍCULA DE CARBÓN

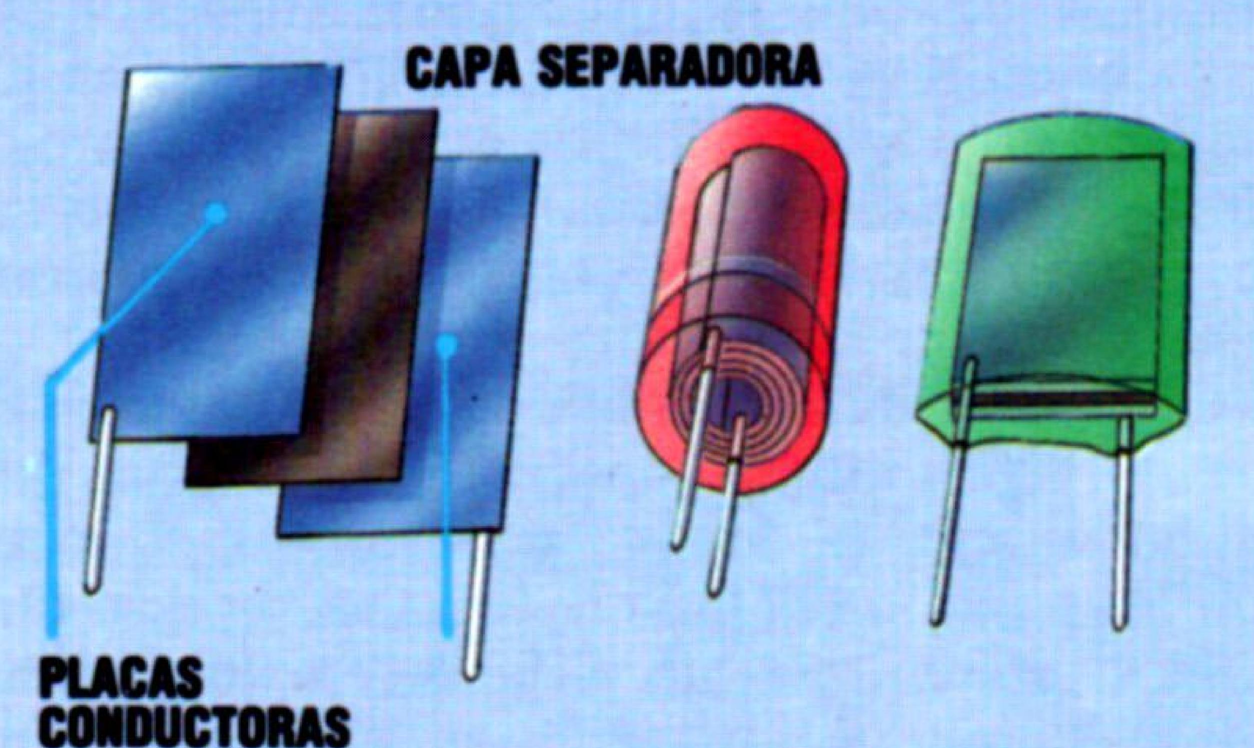
Las resistencias son una de las piezas electrónicas más sencillas: las más comunes son las de alambre enrollado y las de película de carbón. Las primeras se componen de largos trozos de cable enrollado bien ajustado alrededor de un cilindro aislante y colocado dentro de una cubierta aislante. La corriente fluye a lo largo del cable, lo que causa una reducción de la misma. En las resistencias de película de carbón la corriente pasa a través de un corte en espiral de una película de carbón que hay alrededor del cilindro aislante.

Ambos tipos de resistencia están marcados con bandas de colores para identificar sus valores. Aprender a leer estas bandas es muy sencillo. Las dos primeras representan una cifra para la resistencia expresada en ohmios y la tercera un valor por el que se multiplica esta cifra. La banda final, dorada o plateada, indica la tolerancia del componente y se puede utilizar para señalar en qué sentido se deben leer las otras bandas. Siempre se lee en dirección a la banda dorada o plateada. Para no aprenderse de memoria los colores, use un corto programa como el listado al margen para calcular los valores de las resistencias

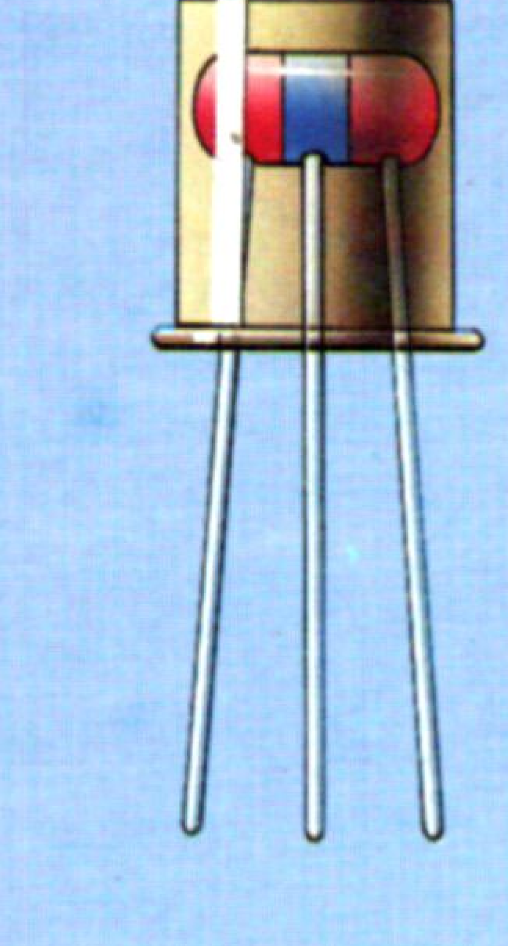


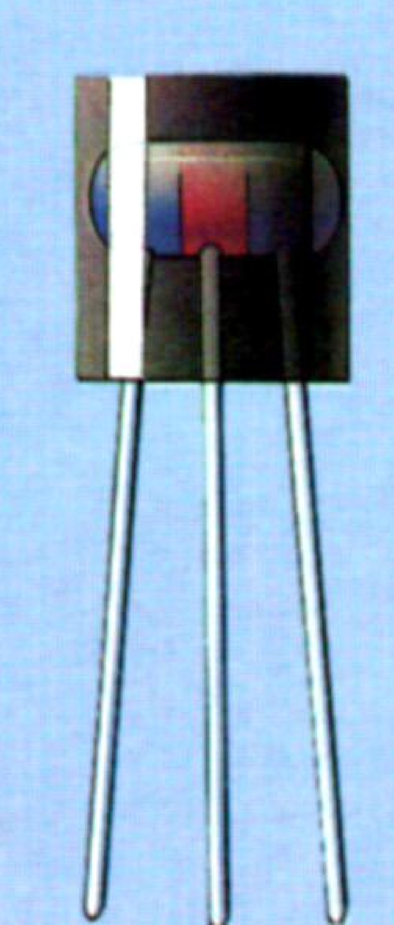
La rueda de las resistencias
Un procedimiento muy rápido para calcular los valores de las resistencias se puede obtener construyendo una rueda con tres discos y un sujetador para papel. Uno simplemente pone en línea los colores y lee los valores

Condensadores



Los condensadores resisten la corriente alterna. Las corrientes CA de alta frecuencia fluyen a través de los condensadores más fácilmente que las de baja frecuencia, lo que resulta útil para limpiar o filtrar una señal eléctrica. Muchos tableros de microprocesadores están llenos de condensadores con este único objetivo. Esencialmente se componen de dos placas conductoras separadas por una capa aislante de varias sales especiales (para voltajes altos) o de cerámica (para voltajes bajos). Las placas pueden ser bastante grandes y las dimensiones reales del componente se reducen enrollando las placas de tal manera que formen una espiral compacta





Transistores

El transistor es la más compleja de estas piezas: es un dispositivo semiconductor y su invención, en 1947, marcó el comienzo de la electrónica moderna. Un transistor posee dos aplicaciones básicas. Como amplificador, puede tomar una pequeña corriente de entrada y producir una gran corriente de salida. Como interruptor, se puede utilizar una corriente para encender o apagar otra corriente. Esta capacidad para actuar como un interruptor electrónico es la base de toda la electrónica digital y es un factor esencial para el funcionamiento de los ordenadores.

Al igual que los diodos, los transistores se componen de materiales semiconductores, pero en su interior poseen dos junturas en lugar de sólo una. Hay tres cables que van hacia los semiconductores, que suelen denominarse base, colector y emisor. Es la forma en que estos tres cables se acoplan al circuito lo que determina si el transistor actúa como un amplificador o como un interruptor. A diferencia de las resistencias y de los diodos, no existen formas estandarizadas para identificar los cables de un transistor ni sus especificaciones. El método usual consiste en consultar el libro de referencias del fabricante para saber el número de ese componente en particular

Kevin Jones

La lozanía de Oric

Esta novel compañía se encuentra en pleno ascenso en el mercado de la informática



Oric Products International se creó para competir con Sinclair Research. Financiada por la British Car Auctions, sus directivos provinieron de Tangerine Computer System y Tansoft.

Cuando se fundó, en 1982, Oric amalgamó la ciencia del marketing (representada por el director gerente Barry Muncaster y el director de ventas Peter Harding) con la del diseño (avalada por Paul Johnson, en hardware, y Paul Kaufman, en software). La máquina creada, el Oric-1, es un equivalente del Spectrum, pero con un 6502. Viene en versiones de 49 Kbytes y 16 Kbytes y, en comparación con el Spectrum, su teclado es mejor, viene en una carcasa más sólida y posee gráficos y sonido superiores. Su BASIC residente es la versión Microsoft estándar de que disponen ordenadores como el Vic-20 y el Apple. Asimismo, el Oric posee una interface para impresora Centronics estándar, que le permite conectar directamente con impresoras de tamaño normal.

Sin embargo, la empresa hubo de hacer frente a una serie de obstáculos importantes, sufriendo demoras en las entregas y fallos de diseño. Las primeras máquinas tenían problemas con la carga de cintas y sobre visualizaciones inestables en pantalla. La ROM de BASIC, con un buen número de errores pequeños pero molestos, hacían que la programación resultara delicada. Además, las técnicas para gráficos del Oric, que economizaban memoria, dificultaban enormemente la programación. Estos problemas se acentuaron cuando la empresa se vio obligada a reajustar el precio de la máquina, situándolo por encima del precio del Spectrum.



La gama Oric
La apariencia futurista del Oric-1 duró quince meses, hasta que se decidió remodelarlo y crear así el elegante Atmos, rojo y negro, con un teclado completo y una ROM de BASIC revisada. La impresora-plotter Oric, a cuatro colores, también se ha vuelto a diseñar para armonizar con la nueva línea

A consecuencia de estas dificultades, al principio el ordenador no se vendió bien y era difícil conseguir software para él. Tansoft trabajó mucho para apoyar a la máquina, proporcionando lenguajes como assembler y FORTH, y es así como actualmente el Oric posee un respetable surtido de software.

Oric también prometió una gama completa de accesorios, incluyendo unidades de disco, un modem y una impresora. De éstos sólo apareció la impresora-plotter a cuatro colores. Después de este tormentoso comienzo, lentamente el Oric ha ido ganando terreno. Las ventas al extranjero arrojan cifras impresionantes. De las 170 000 máquinas que se fabricaron en 1983, más del 50 % se destinaron a la exportación.

Quince meses después de la fecha de su lanzamiento, Oric ya había superado de una forma muy novedosa muchos de los problemas que afloraron con la aparición del Oric 1. Lanzó el modelo Oric Atmos, presentando un teclado completo y unas ROM de BASIC mejoradas; pero en todos los otros aspectos, se trata de la misma máquina remodelada. Con el anuncio de la unidad de disco Hitachi de tres pulgadas para complementar el sistema, el Oric Atmos empieza a parecerse a un micro basado en disco que ofrece una buena relación calidad-precio y en competencia directa con el Acorn Electron y el Commodore 64 más que con el Spectrum.

No obstante, otra vez el optimismo inicial se ha visto rebajado por nuevos problemas. Hubo demoras en los envíos de las unidades de disco y el Atmos tiene ciertas dificultades para cargar y ejecutar las cintas ya existentes. Pero se ha creado un nuevo centro de estudios y proyectos en Cambridge que, junto con un acuerdo comercial con Inmos, el fabricante británico de semiconductores, habrá de proporcionar una sólida base sobre la cual trabajar en el futuro.

Editorial Delta, S.A.



10031

9 788485 822836